Intel® Virtual RAID on CPU (Intel® VROC), Redfish* Representational State Transfer (REST)

API Documentation

Revision 1.3

March 2024

Document Number: 763356

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning Intel products described herein. You agree to grant Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software, or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.

Intel technologies may require enabled hardware, specific software, or services activation. Check with your system manufacturer or retailer.

The products described may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest Intel product specifications and roadmaps.

Copies of documents which have an order number and are referenced in this document may be obtained by calling 1-800-548-4725 or visit www.intel.com/design/literature.htm.

Intel, the Intel logo, are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Contents

Tables

Revision History

Revision Number	Description	Release Date
1.3	•Updated DMTF schema bundle to 2023.1 version. •Changed Volume/Drive Operations.Name to Operations Operation. •Added "HotspareActivationPolicy" property to Storage. •Added "ControllerRates"property to Storage. •Added "HotspareReplacementMode" property to Drive.	March 2024
1.2	•Added "ResetToDefaults" action in Storage. •Added "SecureErase" action in Drive. •Added "DrivePredictiveFailure" event in Event. •Intel® VROC releases mapping. •Supported type of "SecureErase" action. •New "DrivePredictiveFailureCleared" event. •Updated Secure Erase description.	August 2023
1.0	•Initial release of the document.	December 2022

§§

1 Introduction

1.1 Scope

This document is a reference for Intel® Virtual RAID on CPU (VROC) Redfish API.

The API was based on the *Distributed Management Task Force's (DMTF) Redfish Interface Specification and schema* v1.11.1. It is meant to be served by the Intel VROC component acting as a **PLDM RDE device**. Therefore, unless required by the functionality of this API, it does not describe basic resources and actions required by the Redfish server to operate (like Account Service or Sessions). See the official standard documentation for more details. It is highly recommended to read the Redfish standard specification and PLDM RDE standard before this document.

1.2 Intended Audience

The intended audiences for this document are all implementers and users of this Intel VROC Redfish API.

1.3 Terminology

Term	Definition
API	Application Program Interface
BEJ	Binary Encoded JSON
HTTP	Hypertext Transfer Protocol
JSON	JavaScript Object Notation
NVMe	Non-volatile Memory Express
OData	Open Data Protocol
OEM	Original Equipment Manufacturer
PCIe	Peripheral Component Interconnect express
PLDM	Platform Level Data Model
RDE	Redfish Device Enablement
REST	Representational state transfer
SKU	Stock-keeping unit
URI	Uniform Resource Identifier
UUID	Universally Unique Identifier
VROC	Virtual RAID on CPU

1.4 References

Table 1-1. Reference Documents and Resources

Title	Location
Redfish Scalable Platforms Management API Specification v1.14.0	*https://www.dmtf.org/sites/default/files/standards/documents/DSP0266_1.14.0.pdf*
Redfish Base Message Registry	*https://redfish.dmtf.org/registries/Base.1.8.2.json*
Storage Device Message Registry	*https://redfish.dmtf.org/registries/StorageDevice.1.1.0.json*
Platform Level Data Model (PLDM) for Redfish Device Enablement	*https://www.dmtf.org/sites/default/files/standards/documents/DSP0218_1.1.0.pdf*
Platform Level Data Model (PLDM) for Platform Monitoring and Control Specification	*https://www.dmtf.org/sites/default/files/standards/documents/DSP0248_1.2.0.pdf*

1.5 VROC Releases Mapping

VROC Release	VROC Redfish API Document Version
8.0	1.0
8.2 / 8.5	1.2
8.6	1.3

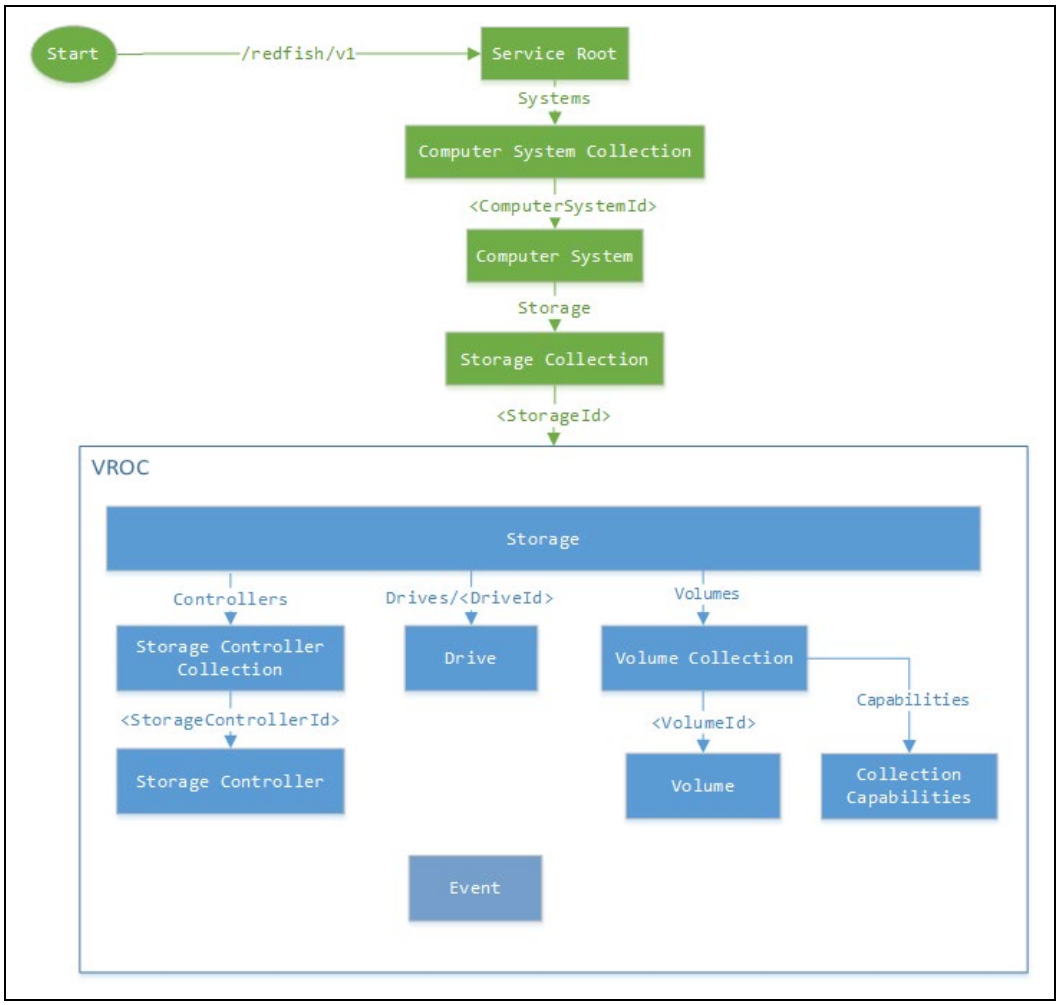
§§

2 VROC Redfish API Model

2.1 Overview

The Intel® VROC Redfish API model uses Redfish schemas to provide RAID level monitoring and management capabilities over managed storage. The below figure shows detailed information on overall API data model.

Table 2-1. VROC Redfish API Model



The diagram above shows all resources returned by the Intel VROC service acting as PLDM RDE Device (shown in the blue square) and their relation to other Redfish resources like Storage Collection. As shown, the Intel VROC service returns nine types of resources: Storage, Storage Controller Collection, Storage Controller, Drive, Event, Volume Collection and Volume (with Volume Collection Capabilities).

Storage resource represents the Intel VROC service software itself, thus there should be exactly one storage resource returned by the Intel VROC service. It aggregates information about managed storage like storage controllers, drives and volumes. Users shall be able to navigate to all Intel VROC resources starting from here, except for Event resources.

Storage Controller Collection contains references to all Storage Controllers managed by Intel VROC. Currently, one storage controller is supported, which manages only NVMe drives (VMD NVMe RAID).

Volume Collection contains references to existing Volumes managed by Intel VROC (VMD NVMe RAID). Using this path, you can create new Volumes. Each created Volume is represented by a Volume schema that contains all properties needed to manage it. Check the Volume Capabilities resource to get more information on required properties for volume creation.

Drive schema represents physical disks attached to the computer system via the Intel VMD Controller, detected and managed by the Intel VROC service. Those drives can be used to create Volumes or act as a spare drive.

2.2 OData fields

All resources returned by the Intel VROC Redfish API support several OData fields. Those fields were omitted in resource description to avoid redundancy. However, they will be shown in response examples. For more details on supported OData fields see the table below.

Table 2-2. Supported OData Fields

OData fields		
@odata.id	string	The unique identifier for a resource in Uri format. For the Intel VROC service, this field will always be returned using BEJ deferred binding, as recommended for RDE Devices (e.g., "%L7").
@odata.type	string	The type of resource (e.g., "#ComputerSystem.v1_4_0.ComputerSystem").
@odata.etag	string	The value of this property shall be a string that is defined by the ETag HTTP header definition in RFC7232.
Members@odata.count	integer	The number of items in a collection. (e.g., "Members@odata.count" : 4)

2.3 BEJ Deferred Binding

The Intel VROC (VMD NVMe RAID) software service acts as a PLDM RDE Device returning data about owned resources in the form of BEJ compliant with Redfish schemas. Some of the properties in Redfish schema require return information outside the scope of PLDM RDE Device control (like full URI path to the resource). For that situation, RDE specification defines so-called 'Deferred binding of data'. It takes the form of macros that RDE Device can use to substitute information it does not have. All macros start with '%' sign and should be substituted by the receiver with actual data.

In this documentation, you can find the usage of the BEJ deferred binding in various examples of data returned by VROC service.

More information about BEJ Deferred Binding can be found in the Platform Level Data Model (PLDM) for Redfish Device Enablement specification (section 8.3)

§§

3 Resources

This section describes all available resources of VROC Redfish API. For a quick overview, the table below lists all VROC Redfish API's available resources and their paths.

Table 3-1. VROC Swordfish API Resources

Resources	URI
Event	N/A
Storage	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}
Storage Controller Collection	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers
Storage Controller	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers/{StorageControllerId}
Drive	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}
Volume Collection	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes
Volume Collection Capabilities	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/Capabilities
Volume	/redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}

Detailed information on the resources and their properties can be found in the following subsections. Several conventions were used when describing those resources:

- Each section contains a table presenting the resource path to which a request should be made and the response object.
- Path params are present in the resource paths and are enclosed in curly brackets (e.g. {SessionId}). In all described requests, they represent the Identifier of the specific resource instance, as suggested, by name.
- For each parameter of the response object, there are four columns showing as follows: parameter's name, type, read-write privileges, and description.
- The third column contains a 'read only' note if the parameter cannot be changed via PATCH request. If the third column is empty, it means that 'read-write' privileges are enabled, and the parameter can be changed via PATCH request.
- 'idRef' type of the parameter is a shortcut for the OData idRef definition (http://redfish.dmtf.org/schemas/v1/odata.v4_0_3.json#/definitions/idRef) that contains the object with the property '@odata.id' and value is a link to the specific resource.
- Each resource section contains corresponding examples of the response from the Intel VROC RDE service. In those examples you can find the usage of BEJ deferred binding substitution parameters.

- The “ANNOTATION” schema (v1.1.1) URI is equal to https://www.intel.com/VROC-redfish-annotations_v1_1_1.

3.1 Event

Table 3-2. Event Resource Properties

N/A			
The Event schema describes the JSON payload received by the client that has subscribed to event notification when events occur. If events pass the criteria of a specific subscription (event destination), it is sent to the address specified by the Event Destination. This Resource contains detailed data about events (v1.8.0).			
Schema URI	https://www.intel.com/VROC-Event_v1_8_0		
Id	string	*read only*	Identifier of the resource.
Name	string	*read only*	The name of the resource.
Description	string	*read only*	The description of this resource.
Events	array	*read only*	This property shall contain an array of Event objects that represent the occurrence of one or more events.
[].EventGroupId	integer	*read only*	The identifier that correlates events with the same root cause. If `0`, no other event is related to this event.
[].EventTimestamp	string	*read only*	The time the event occurred. Shall be set to a sentinel value of zero for RDE Devices not containing real-time clocks (see DSP0248 specification for PLDM RDE).
[].MemberId	string	*read only*	This property uniquely identifies the member within the collection and contain the zero-based array index.
[].Message	string	*read only*	The human-readable event message.
[].MessageArgs	array	*read only*	An array of message arguments that are substituted for the arguments in the message when looked up in the message registry.
[].MessageId	string	*read only*	The identifier for the message. This property shall contain a MessageId, as defined in the 'MessageId format' clause of the Redfish Specification.
[].MessageSeverity	string	*read only*	The severity of the message in this event. Valid values are *OK, Warning* and *Critical.*
[].OriginOfCondition	idRef	*read only*	The reference to the Resource that caused the event to occur.

3.1.1 Supported Events

The below table lists all events supported by the Intel VROC Redfish API with short description of what triggers it. All supported events are based on the Redfish `*Storage*

Device' message registry. For detailed information on returned messages check official registry file (see references section).

NOTE: Intel VROC Redfish API supports only synchronous events. See section 7 for more information.

Table 3-3. Supported events

Message Id	Trigger
StorageDevice.1.0.DriveFailure	Indicates detection of a drive with failure condition (failed state). Either a new failed drive was detected or an existing drive change status to failed.
StorageDevice.1.0.DriveInserted	A new drive has been inserted and detected by the Intel VROC service.
StorageDevice.1.0.DriveRemoved	A drive has been removed (either physically or ejected from the OS).
StorageDevice.1.0.VolumeDegraded	Indicates a volume degraded condition was detected due to one or more drives not installed correctly, missing, or has failed. Volume is in a degraded state. This event uses UUID Identifier of the Volume properties as the id in message args.
StorageDevice.1.0.VolumeFailure	Indicates a RAID volume failure condition was detected due to one or more drives not installed correctly, missing, or has failed. The RAID Volume is in a failed state. This event uses UUID Identifier of the Volume properties as the id in message args.
StorageDevice.1.0.VolumeOK	A volume status has changed to OK. Occurs after successful Volume rebuild.
StorageDevice.1.0.DrivePredictiveFailure	A drive predictive failure state occurred. Drive is predicted to be failed soon.
StorageDevice.1.0.DrivePredictiveFailureCleared	Indicates that previously detected drive predictive failure state does not occur anymore.

3.1.2 Event Examples

3.1.2.1 Drive Failure Event


```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "A failure condition for the drive located in
'Embedded:Port=1C:Box=1:Bay=3' has been detected.",
    "MessageArgs": ["Embedded:Port=1C:Box=1:Bay=3"],
    "MessageId": "StorageDevice.1.0.DriveFailure",
    "MessageSeverity": "Critical",
```

```
    "OriginOfCondition": {
      "@odata.id": "%L6"
    }
  }]
}
```

3.1.2.2 Drive Inserted Event

```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "The drive located in 'Embedded:Port=1C:Box=1:Bay=3' was
inserted.",
    "MessageArgs": ["Embedded:Port=1C:Box=1:Bay=3"],
    "MessageId": "StorageDevice.1.0.DriveInserted",
    "MessageSeverity": "OK",
    "OriginOfCondition": {
      "@odata.id": "%L7"
    }
  }]
}
```

3.1.2.3 Drive Removed Event

```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "The drive located in 'Embedded:Port=1C:Box=1:Bay=3' was
removed.",
    "MessageArgs": ["Embedded:Port=1C:Box=1:Bay=3"],
    "MessageId": "StorageDevice.1.0.DriveRemoved",
    "MessageSeverity": "Critical",
    "OriginOfCondition": {
      "@odata.id": "%L8"
    }
  }]
}
```

3.1.2.4 Volume Degraded Event


```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "The volume 'ebd0bc2d-b286-42fa-b691-5253098a5c1a' attached to
the storage controller located in 'Embedded=NVMe' is degraded.",
    "MessageArgs": ["ebd0bc2d-b286-42fa-b691-5253098a5c1a", "Embedded=NVMe"],
    "MessageId": "StorageDevice.1.0.VolumeDegraded",
    "MessageSeverity": "Warning",
    "OriginOfCondition": {
      "@odata.id": "%L12"
    }
  }]
}
```


3.1.2.5 Volume Failure Event


```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "The volume 'ebd0bc2d-b286-42fa-b691-5253098a5c1a' attached to
the storage controller located in 'Embedded=NVMe' has failed.",
    "MessageArgs": ["ebd0bc2d-b286-42fa-b691-5253098a5c1a", "Embedded=NVMe"],
    "MessageId": "StorageDevice.1.0.VolumeFailure",
    "MessageSeverity": "Critical",
    "OriginOfCondition": {
      "@odata.id": "%L12"
    }
  }]
}
```


3.1.2.6 Volume OK Event


```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
```

```
    "MemberId": "0",
    "Message": "The health of volume 'ebd0bc2d-b286-42fa-b691-5253098a5c1a'
that is attached to the storage controller located in 'Embedded=NVMe' has
changed to OK.",
    "MessageArgs": ["ebd0bc2d-b286-42fa-b691-5253098a5c1a", "Embedded=NVMe"],
    "MessageId": "StorageDevice.1.0.VolumeOK",
    "MessageSeverity": "OK",
    "OriginOfCondition": {
      "@odata.id": "%L12"
    }
  }]
}
```

3.1.2.7 Drive Predictive Failure Event

```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "A predictive failure condition for the drive located in
'Embedded:Port=1C:Box=1:Bay=3' has been detected.",
    "MessageArgs": ["Embedded:Port=1C:Box=1:Bay=3"],
    "MessageId": "StorageDevice.1.0.DrivePredictiveFailure",
    "MessageSeverity": "Critical",
    "OriginOfCondition": {
      "@odata.id": "%L8"
    }
  }]
}
```

3.1.2.8 Drive Predictive Failure Cleared Event

```
{
  "@odata.type": "#Event.v1_7_0.Event",
  "Id": "0",
  "Name": "VROC Event",
  "Description": "Event reported by the Intel VROC service",
  "Events@odata.count": 1,
  "Events": [{
    "EventGroupId": 0,
    "EventTimestamp": 0,
    "MemberId": "0",
    "Message": "A predictive failure condition for the drive
'Embedded:Port=1C:Box=1:Bay=3' has been cleared.",
    "MessageArgs": ["Embedded:Port=1C:Box=1:Bay=3"],
    "MessageId": "StorageDevice.1.0.DrivePredictiveFailureCleared",
    "MessageSeverity": "OK",
    "OriginOfCondition": {
      "@odata.id": "%L8"
    }
```

```
  }]
}
```

3.2 Storage

Table 3-4. Storage resource properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}			
This resource represents the storage subsystem of a specified Computer System. There should be exactly one instance of the Storage resource returned by the Intel VROC service. It represents the storage subsystem managed by Intel VROC (VMD NVMe RAID) and combines information about the Intel VMD Storage Controller, NVMe Drives and Volumes (v1.15.0).			
Schema URI	https://www.intel.com/VROC-Storage_v1_15_0		
Id	string	*read only*	Identifier of the Intel VROC storage subsystem within Storage Collection. Returned as BEJ deferred binding (%I) since the Storage Collection management is outside of the Intel VROC service scope.
Name	string	*read only*	The name of the Storage.
Actions	object	*read only*	The available actions for this resource. Currently the Intel VROC (VMD NVMe RAID) software does not support any actions on this resource.
{}.#Storage.ResetToDefaults	object	*read only*	This action shall reset the storage device to factory defaults. This can cause the loss of data. See **Reset to defaults** section for more details.
Status	object	*read only*	The status and health of the resource and its subordinate or dependent resources.
{}.State	string	*read only*	Represents the current state of the Intel VROC service. Supported values are *Starting* and *Enabled.*
{}.HealthRollup	string	*read only*	This property shall represent the Health State of the resource and its dependent resources (volumes, drives). Supported values are *Ok, Warning* and *Critical.*
HotspareActivationPolicy	string	*read only*	The policy under which hot spare drives managed by Intel VROC will be activated. Only supported value is *OnDriveFailure.*
Drives	array	*read only*	Array of references to the set of drives managed by Intel VROC.
Links	object	*read only*	Contains references to other resources that are related to this resource.
{}.Enclosures	array	*read only*	An array of links to the chassis to which this storage subsystem is attached.
Volumes	idRef	*read only*	A link to the collection of RAID volumes managed by Intel VROC.
Controllers	idRef	*read only*	A link to the collection of Storage Controllers managed by Intel VROC.

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC
```

Response example:


```
{
  "@odata.id": "%L1",
  "@odata.etag": "W/\"5215ea4b\"",
  "@odata.type": "#Storage.v1_11_0.Storage",
  "Id": "%I1",
  "Name": "Intel VROC Storage Subsystem",
  "Actions": {
     "#Storage.ResetToDefaults": {
      "target": "%L1/Actions/Storage.ResetToDefaults",
      "title": "Reset Storage to defaults",
      "ResetType@Redfish.AllowableValues": ["ResetAll", "PreserveVolumes"]
    }
  },
  "Status": {
    "State": "Enabled",
    "HealthRollup": "Ok"
  },
  "Drives@odata.count": 4,
  "Drives": [
    {
      "@odata.id": "%L4"
    },
    {
      "@odata.id": "%L5"
    },
    {
      "@odata.id": "%L6"
    },
    {
      "@odata.id": "%L7"
    }
  ],
  "HotspareActivationPolicy": "OnDriveFailure",
  "Links": {
    "Enclosures@odata.count": 1,
    "Enclosures": [{ "@odata.id": "%C" }]
  },
  "Volumes": {
    "@odata.id": "%L2"
  },
  "Controllers": {
    "@odata.id": "%L3"
  }
}
```

3.3 Storage Controller Collection

Table 3-5. Storage Controller Collection resource properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers			
A collection of references to Storage Controller resource instances.			
Schema URI	https://www.intel.com/VROC-StorageCtrlColl_v1_0_0		
Name	string	*read only*	The name of the resource or array element.
Members	array	*read only*	This property shall contain an array of references to the members of this collection.

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC/Controllers
```

Response example:

```
{
  "@odata.id": "%L3",
  "@odata.etag": "W/\"5215ea4b\"",
  "@odata.type": "#StorageControllerCollection.StorageControllerCollection",
  "Name": "Intel VROC Controller Collection",
  "Members@odata.count": 1,
  "Members": [
    { "@odata.id": "%L15" }
  ]
}
```

3.4 Storage Controller

Table 3-6. Storage Controller Resource Properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers/{StorageControllerId}			
The Storage Controller schema describes a storage controller and its properties. A storage controller represents a physical or virtual storage device that produces volumes (v1.7.0). For the Intel VROC Storage Subsystem there is one Storage Controller resource returned. It represents a virtual controller in form of the Intel VROC (VMD NVMe RAID) software.			
Schema URI	https://www.intel.com/VROC-StorageController_v1_7_0		
Id	string	*read only*	Uniquely identifies the resource.

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers/{StorageControllerId}			
Name	string	*read only*	The VROC specific name of the resource. The following values are expected: - "Volume Management Device Controller" for UEFI agent - "Intel(R) Volume Management Device NVMe RAID Controller" for Windows agent - Linux* format includes SKU information e.g., "Intel(R) VROC (Premium)"
CacheSummary	object	*read only*	The cache memory of the storage controller in general detail.
{}.TotalCacheSizeMiB	integer	*read only*	The total configured cache memory measured in MiB. Equals 0 if cache is not supported.
{}.PersistentCacheSizeMiB	integer	*read only*	The amount of cache memory that is persistent as measured in mebibytes. This size shall be less than or equal to the TotalCacheSizeMib. Equals 0 if cache is not supported.
{}.Status	object	*read only*	The status and health of the resource and its subordinate or dependent resources.
{}.{}.State	string	*read only*	Current state of the cache. Shall be set to *Absent* since VROC does not provide cache support.
{}.{}.Health	string	*read only*	The health state of cache. Valid values are *Ok, Warning* and *Critical.*
FirmwareVersion	string	*read only*	Overall version of the Intel VROC service.
Identifiers	array	*read only*	The array of durable names for the storage controller.
[].DurableName	string	*read only*	The world-wide, persistent name of the resource.
[].DurableNameFormat	string	*read only*	The format of the durable name property. Valid values are *NAA*, *iQN*, *FC_WWN*, *UUID*, *EUI*, *NQN*, *NSID*.
Location	object	*read only*	The location of the Intel VMD storage controller.
{}.PartLocation	object	*read only*	The part location within the placement.
{}.{}.ServiceLabel	string	*read only*	The label of the part location, such as a silk-screened name or a printed label.
{}.{}.LocationType	string	*read only*	The type of location of the part. For VROC Storage Controller it should always be *Embedded.*
Manufacturer	string	*read only*	The manufacturer of the Intel VMD storage controller.
Model	string	*read only*	The model number for the Intel VMD storage controller.

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers/{StorageControllerId}			
SKU	string	*read only*	Represents type of Intel VROC (VMD NVMe RAID) Software Key. Valid values are: *'Pass-Thru', 'Pass-Thru-Trial', 'Intel-SSD-Only', 'Standard'* and *'Premium'*. For more information on VROC Software Keys see section 6.4.
Status	object	*read only*	The status and health of the resource and its subordinate or dependent resources.
{}.State	string	*read only*	The known state of the resource, such as *Enabled*.
{}.Health	string	*read only*	The health state of the resource. Valid values are *Ok, Warning* and *Critical.*
SupportedControllerProtocols	array	*read only*	The supported set of protocols for communicating to this Intel VMD storage controller. Supported value is *PCIe.*
SupportedDeviceProtocols	array	*read only*	The protocols that the Intel VMD storage controller can use to communicate with attached devices. Supported values are *NVMe*.
SupportedRAIDTypes	array	*read only*	This object describes the RAID Types supported by the Intel VMD storage controller. Valid values are *RAID0, RAID10, RAID1, RAID5.*
ControllerRates	array		This object describes the various controller rates used for processes such as volume rebuild, transformation or consistency checks.

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC/Controllers/1
```

Response example:


```
{
  "@odata.id": "%L15",
  "@odata.etag": "W/\"5215ea4b\"",
  "@odata.type": "#StorageController.v1_3_0.StorageController",
  "Id": "1",
  "Name": "Intel(R) VROC (Premium)",
  "CacheSummary": {
    "TotalCacheSizeMiB": 0,
    "PersistentCacheSizeMiB": 0,
    "Status": {
      "State": "Absent",
      "Health": "Ok"
    }
  },
  "FirmwareVersion": "8.6.0.1023",
  "Identifiers": [],
  "Location": {
    "PartLocation": {
      "ServiceLabel": "Embedded=NVMe",
      "LocationType": "Embedded"
```

```
    }
  },
  "Manufacturer": "Intel",
  "Model": "Intel VROC 8.6",
  "SKU": "Premium",
  "Status": {
    "State": "Enabled",
    "Health": "Ok"
  },
  "SupportedControllerProtocols": ["PCIe"],
  "SupportedDeviceProtocols": ["NVMe"],
  "SupportedRAIDTypes": ["RAID0", “RAID1”, “RAID5”, “RAID10”],
  “ControllerRates”: [
   {
     "ConsistencyCheckRatePercent": 50
   },
   {
     "RebuildRatePercent": 50
   },
   {
     "TransformationRatePercent": 50
   }
  ]
}
```

3.5 Drive

Table 3-7. Drive resource properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
The Drive schema represents a single physical disk drive for a system, including links to associated Volumes (v1.17.0).			
Schema URI	https://www.intel.com/VROC-Drive_v1_17_0		
Id	string	*read only*	Uniquely identifies the resource.
Name	string	*read only*	The name of the Drive. The expected format is: “[Advertised Drive Capacity]GB(or TB) [Protocol] [MediaType]” e.g., "375GB NVMe SSD".
Actions	object	*read only*	The available actions for this resource.
{}.#Drive.SecureErase	object	*read only*	This action is used to securely erase the contents of the drive.
Status	object	*read only*	Object describing drive overall status.
{}.State	string	*read only*	Current state of the drive. Supported values are *Enabled, Disabled, StandbyOffline, StandbySpare, UnavailableOffline, Updating* and *Absent.*

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
{}.Health	string	*read only*	The health state of the drive. Valid values are *Ok, Warning* and *Critical.*
BlockSizeBytes	integer	*read only*	The size of the smallest addressable unit (Block) of this drive, in bytes.
CapableSpeedGbs	number	*read only*	The speed, in gigabit per second (Gbit/s), at which this drive can communicate to a storage controller in ideal conditions.
CapacityBytes	integer	*read only*	Drive's total size in bytes.
EncryptionAbility	string	*read only*	The encryption ability of the drive. Supported values are: *None* and *SelfEncryptedDrive.*
EncryptionStatus	string	*read only*	The encryption status of the drive. Supported values are: *Locked, Unencrypted* and *Unlocked.*
FailurePredicted	boolean	*read only*	This property shall indicate whether this drive currently predicts a manufacturer-defined failure. In the Intel VROC software, it reflects the Critical *SMART events on drive (At Risk).*
FirmwareVersion	string	*read only*	Contains the currently active firmware revision of the drive.
HotspareReplacementMode	string	*read only*	The replacement mode for the hot spare drive. Valid value is NonRevertible.
HotspareType	string		The type of hotspare this drive is currently serving as. Valid values for VROC Swordfish API service are "None" when drive is not a hotspare and "Global" when drive serves as a hotspare for all Volumes of Storage System.
Identifiers	array	*read only*	The array of durable names for the drive.
[].DurableName	string	*read only*	The world-wide, persistent name of the resource.
[].DurableNameFormat	string	*read only*	The format of the durable name property. Valid values are *NAA*, *iQN*, *FC_WWN*, *UUID*, *EUI*, *NQN*, *NSID*.
LocationIndicatorActive	boolean		An indicator allowing an operator to physically locate this resource. For more information on managing LEDs, see section **Activate drive LED**.
Links	object	*read only*	Contains references to other resources that are related to this resource.
{}.Volumes	array	*read only*	Array containing reference to Volumes the drive is part of if any.
{}.Storage	object	*read only*	The link to the storage subsystem to which this drive belongs.
{}.Chassis	object	*read only*	The link to the chassis that contains the drive.

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
MediaType	string	*read only*	The type of media contained in this drive. Valid values are *HDD* and *SSD.*
Model	string	*read only*	Model number of the drive.
NegotiatedSpeedGbs	number	*read only*	The speed, in gigabit per second (Gbit/s), at which this drive currently communicates to the storage controller.
Operations	array	*read only*	The operations currently running on the Drive. Empty if no operation is running.
[].Operation	string	*read only*	The name of the operation.
[].PercentageComplete	integer	*read only*	The percentage of the operation that has been completed.
PhysicalLocation	object	*read only*	The location of the drive.
{}.PartLocation	object	*read only*	The part location within the placement.
{}.{}.ServiceLabel	string	*read only*	The label of the part location, such as a silk-screened name or a printed label.
{}.{}.LocationType	string	*read only*	The type of location of the part. Valid values are *Slot, Bay, Connector* and *Socket*.
{}.{}.LocationOrdinalValue	integer	*read only*	This property shall contain the number that represents the location of the part based on the LocationType.
Protocol	string	*read only*	The protocol the drive is using to communicate to the storage controller. Supported values is *NVMe*.
PredictedMediaLifeLeftPercent	integer	*read only*	The percentage of read and writes that are predicted to be available for the drive.
Revision	string	*read only*	Contains the currently active firmware revision of the drive.
SerialNumber	string	*read only*	Contains the serial number of the drive that is assigned by the vendor.
StatusIndicator	string		The state of the status indicator, which communicates status information about this drive. Supported values are *OK, Rebuild, Fail, Hotspare, InACriticalArray* and *InAFailedArray.*
WriteCacheEnabled	boolean		Indicates if write cache is enabled for the drive.

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC/Drives/1234567890A
```

Response example:


```
{
  "@odata.id": "%L6",
  "@odata.etag": "W/\"5215ea4b\"",
  "@odata.type": "#Drive.v1_13_0.Drive",
  "Id": "1234567890A",
  "Name": "500GB NVMe SSD",
  "Actions": {
    "#Drive.SecureErase ": {
      "target": "%L6/Actions/Drive.SecureErase",
      "title": "Secure Erase"
    },
  "Status": {
    "State": "Enabled",
    "Health": "Ok"
  },
  "BlockSizeBytes": 512,
  "CapableSpeedGbs": 64,
  "CapacityBytes": 429496729600,
  "EncryptionAbility": "SelfEncryptedDrive",
  "EncryptionStatus": "Locked",
  "FailurePredicted": false,
  “FirmwareVersion”: “VAAAZ292”
  "HotspareType": "None",
  “HotspareReplacementMode”: "NonRevertible",
  "Identifiers": [
    {
      "DurableName": "00:25:38:6b:61:00:46:b6",
      "DurableNameFormat": "EUI"
    },
    {
      "DurableName": "c1422bee-0586-5569-aeb3-966e3ecedff8",
      "DurableNameFormat": "UUID"
    }
  ],
  "LocationIndicatorActive": false,
  "Links": {
    "Volumes@odata.count": 1,
    "Volumes": [{"@odata.id":"%L22"}],
    "Chassis": {"@odata.id":"%C"},
    "Storage": {"@odata.id":"%L1"}
  },
  "MediaType": "SSD",
  "Manufacturer": "Intel",
  "Model": "INTEL SSDPE2KX010T8",
  "NegotiatedSpeedGbs": 64,
  "Operations": [],
  "PhysicalLocation": {
    "PartLocation": {
      "ServiceLabel": "Embedded:Port=1C:Box=1:Bay=3",
      "LocationType": "Bay",
      "LocationOrdinalValue": 3
    }
  },
  "Protocol": "NVMe",
  "Revision": "VAAAZ292",
```

```
  "SerialNumber": "BTLJ722608681P0FGN",
  "StatusIndicator": "OK",
  "PredictedMediaLifeLeftPercent": 95,
  "WriteCacheEnabled": true
}
```

3.6 Volume Collection

Table 3-8. Volume Collection Resource Properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes			
A collection of references to Volume resource instances.			
Schema URI	https://www.intel.com/VROC-VolumeColl_v1_0_0		
Name	string	*read only*	The name of the resource or array element.
Members	array	*read only*	This property shall contain an array of references to the members of this collection.
@Redfish.CollectionCapabilities	object	*read only*	Annotation object representing the POST capabilities of a collection.
{}.Capabilities	array	*read only*	The list of capabilities supported by this resource. Each object in an array describes a capability of a collection for a specific use case.
{}.[].CapabilitiesObject	idRef	*read only*	The link to the resource the client can issue a GET request against to understand how to form a POST request for a collection.
{}.[].UseCase	string	*read only*	The use case in which a client can issue a POST request to the collection. The only supported use case is *VolumeCreation*.
{}.[].Links	object	*read only*	The links to other resources that are related to this resource.
{}.[].{}.TargetCollection	idRef	*read only*	The link to the collection that this capability's structure is describing.
{}.MaxMembers	integer	*read only*	The maximum number of members allowed in this collection.

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes
```

Response example:

```
{
  "@odata.id": "%L21",
  "@odata.etag": "W/\"5215ea4b\"",
  "@odata.type": "#VolumeCollection.VolumeCollection",
```

```
  "Name": "Intel VROC Volume Collection",
  "Members@odata.count": 2,
  "Members": [
    { "@odata.id": "%L22" },
    { "@odata.id": "%L23" }
  ],
  "@Redfish.CollectionCapabilities": {
    "@odata.type": "#CollectionCapabilities.v1_2_2.CollectionCapabilities",
    "Capabilities": [{
      "CapabilitiesObject": {
        "@odata.id": "%L37"
      },
      "UseCase": "VolumeCreation",
      "Links": {
        "TargetCollection": {
          "@odata.id": "%L21"
        }
      }
    }],
    "MaxMembers": 24
  }
}
```


3.7 Volume Collection Capabilities

Table 3-9. Volume Collection Capabilities Resource Properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/Capabilities			
A resource describing the properties allowed in the POST request sent to the Volume Collection to create a new Volume. This resource follows the same schema for the resource that the resource collection contains. It contains annotations to show which properties the client can use in the POST request body (Volume.v1_9_0).			
Schema URI	https://www.intel.com/VROC-Volume_v1_9_0		
Id	string	*read only*	The id of the resource.
Name	array	*read only*	This name of the resource
<PropertyName>@Redfish.RequiredOnCreate	boolean	*read only*	Required in the POST request body.
<PropertyName>@Redfish.OptionalOnCreate	boolean	*read only*	Not required in the POST request body.
<PropertyName>@Redfish.SetOnlyOnCreate	boolean	*read only*	Cannot be modified after the resource is created.
<PropertyName>@Redfish.UpdatableAfterCreate	boolean	*read only*	Can be modified after the resource is created.

<PropertyName>@Redfish.AllowableValues	Array (string)	*read only*	Can be set to any of the listed values.

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/Capabilities
```

Response example:

```
{
  "@odata.id": "%L44",
  "@odata.etag": "W\"5215ea4b\"",
  "@odata.type": "#Volume.v1_9_0.Volume",
  "Id": "Capabilities",
  "Name": "Capabilities for the volume collection",
  "Name@Redfish.OptionalsOnCreate": true,
  "DisplayName@Redfish.UpdatableAfterCreate": true,
  "RAIDType@Redfish.RequiredOnCreate": true,
  "RAIDType@Redfish.AllowableValues": ["RAID0", "RAID1", "RAID10", "RAID5"],
  "CapacityBytes@Redfish.OptionalOnCreate": true,
  "CapacityBytes@Redfish.UpdatableAfterCreate": true,
  "InitializeMethod@Redfish.OptionalOnCreate": true,
  "InitializeMethod@Redfish.AllowableValues": ["Background", "Skip"],
  "StripSizeBytes@Redfish.OptionalOnCreate": true,
  "WriteHoleProtectionPolicy@Redfish.OptionalOnCreate": true,
  "WriteHoleProtectionPolicy@Redfish.UpdatableAfterCreate": true,
  "WriteHoleProtectionPolicy@Redfish.AllowableValues":  ["Off",  "DistributedLog",
“Journaling”],
  "Links@Redfish.RequiredOnCreate": true,
  "Links": {
    "Drives@Redfish.RequiredOnCreate": true,
    "JournalingMedia@Redfish.OptionalOnCreate": true,
    "JournalingMedia@Redfish.UpdatableAfterCreate": true
  },
}
```

3.8 Volume

Table 3-10. Volume Resource Properties

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
Represents Volume managed by the Intel VROC service. It’s the storage area on two or more drives whose RAID type dictates the configuration of the data stored (v1.9.0).			
Schema URI	https://www.intel.com/VROC-Volume_v1_9_0		
Response parameters (Object)			
Id	string	*read only*	Identifier of the volume.

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
Name	string	*read only*	The name of the volume. Equivalent of DisplayName property. Maximum supported length of the property is **16** characters.
Actions	object	*read only*	The available actions for this resource.
{}.#Volume.ChangeRAIDLayout	object	*read only*	This action is used to request system change the RAID layout of the volume based on the provided parameters. See **Change RAID layout** section for more details.
{}.#Volume.CheckConsistency	object	*read only*	This action is used to force a check of the Volume's parity or redundant data to ensure it matches calculated values. See **Check volume consistency** section for more details.
{}.#Volume.ForceEnable	object	*read only*	This action is used to request system force the volume to an enabled state regardless of data loss. See **Force volume to enabled state** section for more details.
{}.#Volume.Initialize	object	*read only*	This action is used to prepare the contents of the volume for use by the system. This action has one optional *parameter* named **InitializeMethod** (the same as property on Volume). This parameter reflects desired method to be used for volume initialization. The type of the method used in the end by the service should be assigned to the Volume property InitializeMethod upon successful initialization. See **Initialize volume** and **RAID volume initialization methods** sections for more details on action request. **NOTE:** On Linux** operating system, this action is ***not supported*** and will return error for all created Volumes. **NOTE:** For RAID0, this action is ***not supported*** and will return error for all created Volumes.
Status	object	*read only*	Object describing volume overall status.
{}.State	string	*read only*	Current state of the volume. Supported values are *Enabled, StandbyOffline, UnavailableOffline* and *Updating*.

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
{}.Health	string	*read only*	The health state of the volume. Valid values are *Ok, Warning* and *Critical.*
BlockSizeBytes	integer	*read only*	The size of the smallest logical addressable unit (Block) of this drive, in bytes.
CapacityBytes	integer		Volume's total size in bytes. If set to '0' when creating volume, the service will set this property's value to all space available on the selected drives.
DisplayName	string		User settable name of the volume. Writable equivalent of Name property. Maximum supported length of the property is **16** characters.
Identifiers	array	*read only*	The array of durable names for the volume.
[].DurableName	string	*read only*	The world-wide, persistent name of the resource.
[].DurableNameFormat	string	*read only*	The format of the durable name property. Valid values are *NAA*, *iQN*, *FC_WWN*, *UUID*, *EUI*, *NQN*, *NSID*.
InitializeMethod	string	*read only*	This value reflects the most recently used Initialization Method (either during Volume creation or Initialize action). Supported values are '*Background'* and *'Skip'*. **NOTE:** For RAID0, only 'Skip' method is supported.
Links	object	*read only*	Contains references to other resources that are related to this resource.
{}.Drives	array	*read only*	An array of references to the drives which contain this volume. This will reference Drives that either wholly or only partly contain this volume.
{}.DedicatedSpareDrives	array	*read only*	An array of references to the drives which are dedicated spares for this volume.
{}.JournalingMedia	idRef		This shall be a pointer to the journaling media used for this Volume to address the RAID write hole issue. Valid when WriteHoleProtectionPolicy property is set to 'Journaling'.
MediaSpanCount	integer	*read only*	Indicates the number of media elements used per span in the secondary RAID for a hierarchical RAID type. Valid only for RAID10.

[GET] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
Operations	array	*read only*	The operations currently running on the Volume. Empty if no operation is running. Supported operations: *Initializing, Rebuilding, Transforming, Verifying*.
[].Operation	string	*read only*	The name of the operation.
[].PercentageComplete	integer	*read only*	The percentage of the operation that has been completed.
OptimumIOSizeBytes	integer	*read only*	The size in bytes of this Volume's optimum IO size for writes and reads operations.
RAIDType	string	*read only*	The RAID type of this volume. Supported values are: *RAID0, RAID1, RAID10, RAID5.*
ReadCachePolicy	string	*read only*	Indicates the read cache policy setting for the Volume. Currently not supported by VROC, will always be 'Off'.
WriteCachePolicy	string	*read only*	Indicates the write cache policy setting for the Volume. Currently not supported by VROC, will always be 'Off'.
StripSizeBytes	integer	*read only*	Size of the strips on each drive in this volume, in bytes.
VolumeUsage	string	*read only*	Indicates the Volume usage type setting for the Volume. Supported values are "Data" and "SystemData" (the volume contains system partition that is used to boot the system).
WriteHoleProtectionPolicy	string		The policy that the RAID volume is using to address the write hole issue. By default, set to "Off". Valid values are: *Off, DistributedLog, Journaling.*

Request example:

```
GET https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/1
```

Response example:

```
{
  "@odata.id": "%L22",
  "@odata.etag": "W\"5214ea4b\"",
  "@odata.type": "#Volume.v1_6_2.Volume",
  "Id": "1",
  "Name": "Volume_001",
  "Actions": {
    "#Volume.ChangeRAIDLayout": {
```

```
      "target": "%L22/Actions/Volume.ChangeRAIDLayout",
      "title": "Change RAID Layout"
    },
    "#Volume.CheckConsistency": {
      "target": "%L22/Actions/Volume.CheckConsistency",
      "title": "Check consistency"
    },
    "#Volume.ForceEnable": {
      "target": "%L22/Actions/Volume.ForceEnable",
      "title": "Force Enable"
    },
    "#Volume.Initialize": {
      "target": "%L22/Actions/Volume.Initialize",
      "title": "Initialize volume",
      "InitializeMethod@Redfish.AllowableValues": ["Background"]
    }
  },
  "Status": {
    "State": "Updating",
    "Health": "Warning"
  },
  "BlockSizeBytes": 512,
  "CapacityBytes": 429496729600,
  "DisplayName": "Volume_001",
  "Identifiers": [
    {
      "DurableName": "ebd0bc2d-b286-42fa-b691-5253098a5c1a",
      "DurableNameFormat": "UUID"
    }
  ],
  "InitializeMethod": "Background",
  "Links": {
    "Drives@odata.count": 3,
    "Drives": [
      {
        "@odata.id": "%L4"
      },
      {
        "@odata.id": "%L5"
      },
      {
        "@odata.id": "%L6"
      }
    ],
    "DedicatedSpareDrives@odata.count": 0,
    "DedicatedSpareDrives": []
  },
  "Operations": [{
      "Operation": "Rebuild",
      "PercentageComplete": 24
  }],
  "RAIDType": "RAID5",
  "ReadCachePolicy": "Off",
  "WriteCachePolicy": "Off",
  "StripSizeBytes": 65536,
  "OptimumIOSizeBytes": 32768,
  "VolumeUsage": "Data",
```

```
  "WriteHoleProtectionPolicy": "Off"
}
```

§§

4 Operations

This section describes operations supported by VROC Redfish API, as per specific objects, excluding GETs covered in the previous chapter.

4.1 Storage

4.1.1 Reset to Defaults

Table 4-1. Reset to Defaults Action

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Actions/Storage.ResetToDefaults			
The reset action resets the storage device to factory defaults. This can cause the loss of data. **NOTE:** When system is booted to operating system installed on the RAID volume, invoking the operation with *ResetType = ResetAll,* will result with no action taken (no data will be removed).			
Request parameters (object)			
ResetType	string	*required*	This parameter shall contain the type of reset to defaults. Valid values are: *ResetAll, PreserveVolumes.* Type *ResetAll* removes all RAID volumes and reset all drives behind VMD to pass-through. Type *PreserveVolume* takes no action.

Request example:

```
POST
https://example.com/redfish/v1/Systems/Host/Storage/VROC/Actions/Storage.ResetToDefaults
{
  "ResetType": "ResetAll"
}
```

4.2 Storage Controller

4.2.1 Set Controller Rate

Table 4-2. Set Controller Rate Operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Controllers/{StorageControllerId}
Operation used to set various controller rate used for processes such as volume rebuild (RebuildRatePercent), transformation (TransformationRatePercent) or consistency checks (ConsistencyCheckRatePercent). The higher rate is set, the higher priority given operation has over other non-RAID system operations.

NOTE: Supported only on Linux* operating system. **NOTE:** Setting one controller rate will affect the remaining as well. Only one controller rate can be set within one request.			
Request parameters (object)			
ControllerRates	array	*required*	Array of different controller rates to be set: ConsistencyCheckRatePercent to set Consistency Check rate to be used during RAID volume *Verify* operation. RebuildRatePercent to set Rebuild rate to be used during RAID volume *Rebuilding* operation. TransformationRatePercent to set Transformation rate to be used during RAID volume *Transforming* operation.

Request example:

```
PATCH https://example.com/redfish/v1/Systems/0/Storage/VROC/Controllers/0
{
  "ControllerRates": [
    {
      "ConsistencyCheckRatePercent": 50
    }
  ]
}
```

4.3 Drive

4.3.1 Activate Drive LED

Table 4-3. Activate Drive LED Operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
Operation used to request activation of the hardware-specific Light-emitting diode (LED) of a specified drive to locate that drive. **NOTE:** On Windows operating system, this action behaves differently from other OSes. After activating the LED with PATCH request, it will light up and turn off automatically after set time.			
Request parameters (object)			
LocationIndicatorActive	boolean	*required*	An indicator allowing an operator to physically locate this resource. If set to *true* means that service should activate drive location LED.

Request example:

```
PATCH https://example.com/redfish/v1/Systems/0/Storage/VROC/Drives/0
{
  "LocationIndicatorActive": true
}
```

4.3.2 Mark Drive as Normal

Table 4-4. Mark drive as normal operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
Operation used to reset a failed drive back to Normal state.			
Request parameters (object)			
StatusIndicator	string	required	The state of the status indicator, which communicates status information about this drive. Shall be set to OK.

Request example:

```
PATCH https://example.com/redfish/v1/Systems/0/Storage/VROC/Drives/0
{
  "StatusIndicator": "OK"
}
```

4.3.3 Mark/Unmark Drive as Spare

Table 4-5. Mark/Unmark Drive as Spare Operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
Operation used to switch specified drive usage between spare and non-spare drive (pass-through).			
Request parameters (object)			
HotspareType	string	*required*	The type of spare this drive should be if any. To **mark drive as spare** set it to *Global*. To **unmark spare drive** and reset it to pass-through set it to *None*.

Request example – mark drive as spare:

```
PATCH https://example.com/redfish/v1/Systems/0/Storage/VROC/Drives/0
{
  "HotspareType": "Global"
}
```

Request example – unmark spare drive:

```
PATCH https://example.com/redfish/v1/Systems/0/Storage/VROC/Drives/0
{
  "HotspareType": "None"
}
```

4.3.4 Set Write Cache on Drive

Table 4-6. Set write cache on drive operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}			
Operation used to update drive write cache state.			
Request parameters (object)			
WriteCacheEnabled	boolean	*required*	New value for this property to be set.

Request example:

```
PATCH
https://example.com/redfish/v1/Systems/host/Storage/VROC/Drives/123456789
{
  "WriteCacheEnabled": true
}
```

4.3.5 Secure Erase

Table 4-7 Secure Erase Action

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Drives/{DriveId}/Actions/SecureErase
This action is used to securely erase the drive content by performing Crypto Erase sanitize operation. No other operation types are currently supported. **NOTE:** Secure erase action is not supported on drives containing VROC metadata (e.g., RAID volume member, spare drive). **NOTE:** No dedicated event is generated once sanitize operation has been finished. To ensure operation has been completed, the user should refresh / reload drive's partition table and verify contents of the drive has been wiped out.

Request example:

```
POST https://example.com/redfish/v1/Systems/Host/Storage/VROC/Drives/0/Actions/Drive.SecureErase
```

4.4 Volume

4.4.1 Change RAID Layout

Table 4-8. Change RAID Layout Action

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}/Actions/Volume.ChangeRAIDLayout			
Action used to initiate change of a volume RAID Layout based on given parameters. Successful execution of this request might start volume migration by the Intel VROC service. From that point, information on the status and progress can be seen under Operations property in Volume resource. **NOTE:** On Linux* operating system only one property can be changed in one request unless changing RAIDType value requires adding new drives to Drives property. **NOTE:** Changing StripSizeBytes requires changing RAIDType in one request on Windows operating system.			
Request parameters (object)			
StripSizeBytes	integer	optional	The new strip size on each drive of the volume.
RAIDType	string	optional	The new RAID Type of the volume.
Drives	array	required	Array of references to the drives that should be added to selected volume. Should be empty if no drives to add.

Request example:

```
POST
https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0/Actions/Volume.ChangeRAIDLayout
{
  "StripSizeBytes": 16384,
  "RAIDType": "RAID10",
  "Drives": [
    {
      "@odata.id": "%L6"
    }
  ]
}
```

4.4.2 Check volume Consistency

Table 4-9. Check Volume Consistency Action

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}/Actions/Volume.CheckConsistency
Action used to initiate the verify process, which looks for bad blocks and conflicts in redundant information. Successful execution of this request means that operation was started by the Intel VROC service. From that point, information on the status and progress can be seen under the *Operations* property in the Volume resource.

Request example:

```
POST https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0/Actions/Volume.CheckConsistency
```

4.4.3 Delete Volume

Table 4-10. Delete Volume Operation

[DELETE] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}
Operation used to delete the volume.

Request example:

```
DELETE https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0
```

4.4.4 Expand Volume

Table 4-11. Expand Volume Operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
Operation used to initiate a volume capacity expansion. Successful execution of this request means that expansion of a volume was started by the Intel VROC service. From that point, information on the status and progress can be seen under the *Operations* property in the Volume resource. **NOTE:** Switching OSs (between Linux* and Windows) "while" a Volume is being expanded is not supported.			
Request parameters (object)			
CapacityBytes	integer	*required*	The new size of volume in Bytes. If equals to 0, then all available array space will be used.

Request example:

```
PATCH https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0
{
  "CapacityBytes": 0
}
```

4.4.5 Force Volume to Enabled State

Table 4-12. Force Enable Action

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}/Actions/Volume.ForceEnable
This action shall request the system to force the volume to enabled state regardless of data loss scenarios.

Request example:

```
POST https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0/Actions/Volume.ForceEnable
```

4.4.6 Initialize Volume

Table 4-13. Initialize Volume Action

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}/Actions/Volume.Initialize			
Action used to start the initialization process on the redundant RAID volume (adding parity / mirroring). **NOTE:** On Linux* operating system, this action is not supported and will return error for all created Volumes. **NOTE:** For RAID0, this action is not supported and will return error for all created Volumes.			
Request parameters (object)			
InitializeMethod	string	*optional*	The method of initialization to be performed. The only supported method is 'Background'.

Request example:

```
POST https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0/Actions/Volume.Initialize
{
  "InitializeMethod": "Background"
}
```

4.4.7 Rename Volume

Table 4-14. Rename Volume Operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
Operation used to rename the volume.			
Request parameters (object)			
DisplayName	string	*required*	The new name of the volume.

Request example:

```
PATCH https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0
{
  "DisplayName": "Renamed volume"
}
```

4.4.8 Set RAID Write Hole Protection Policy

Table 4-15. Set RAID Write Hole Protection Policy Operation

[PATCH] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes/{VolumeId}			
Operation used to change RAID Write Hole Protection Policy. **NOTE:** On Linux* operating system, the "Journaling" policy is not supported.			
Request parameters (object)			
WriteHoleProtectionPolicy	string	*required*	New value for this property to be set.
Links	object	*optional*	Contains references to other resources that are related to this resource.
{}.JournalingMedia	string	*optional*	This shall be a pointer to the journaling media used for this Volume to address the RAID write hole issue. Required when WriteHoleProtectionPolicy property is set to 'Journaling'. Intel VROC (VMD NVMe RAID) supports only Drives as a Journaling Media resource.

Request example:

```
PATCH https://example.com/redfish/v1/Systems/Host/Storage/VROC/Volumes/0
{
  "WriteHoleProtectionPolicy": "Journaling",
  "Links": {
    "JournalingMedia": {
      "@odata.id": "%L5"
    }
  }
}
```

4.5 Volume Collection

4.5.1 Create Volume

Table 4-16. Create Volume Operation

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes			
Operation used to create new volume.			
Request parameters (object)			
Name	string	*optional*	The name of the volume to be created. Equivalent of DisplayName property. Maximum supported length of the property is **16** characters.

[POST] /redfish/v1/Systems/{ComputerSystemId}/Storage/{StorageId}/Volumes			
RAIDType	string	*required*	RAID type of this volume.
CapacityBytes	integer	*optional*	Volume's total size in bytes. If set to **'0'**, the service will set this property's value to all space available on the selected drives. Recommended minimal Volume size is **100MiB** *(1049000 bytes).*
InitializeMethod	string	*optional*	The method of initialization to be performed. The supported methods are 'Skip' and 'Background'. **NOTE:** For RAID0, only *'Skip'* method is supported.
StripSizeBytes	integer	*optional*	Size of the strips on each drive in this volume, in bytes. Supported values are: For **RAID0:** *4096, 8192, 16384, 32768, 65536, 131072* For **RAID1:** *65536* For **RAID5:** *16384, 32768, 65536, 131072* For **RAID10:** *4096, 8192, 16384, 32768, 65536*
WriteHoleProtectionPolicy	string	*optional*	The policy that the RAID volume is using to address the write hole issue. Required for RAID5 type of RAID. By default, set to "Off". Valid values are: *Off, DistributedLog, Journaling.* **NOTE:** *Journaling* policy is **not** supported on Linux* operating system.
Links	object	*required*	Contains references to other resources that are related to this resource.
{}.Drives	array	*required*	An array of references to the drives that will be used to create new volume.
{}.JournalingMedia	string	*optional*	This shall be a pointer to the journaling media used for this Volume to address the RAID write hole issue. Required when WriteHoleProtectionPolicy property is set to 'Journaling'. VROC supports only Drives as a Journaling Media resource. **NOTE:** Journaling Media is **not** supported on Linux* operating system.

Request example:

```
POST https://example.com/redfish/v1/StorageServices/VROC/Volumes
{
  "Name": "Volume_001",
  "RAIDType": "RAID5",
  "CapacityBytes": 429496729600,
  "InitializeMethod": "Background",
  "StripSizeBytes": 8192,
```

```
  "WriteHoleProtectionPolicy": "Journaling",
  "Links": {
    "Drives": [
      {
        "@odata.id": "%L4"
      },
      {
        "@odata.id": "%L5"
      },
      {
        "@odata.id": "%L6"
      }
    ],
    "JournalingMedia": {
      "@odata.id": "%L7"
    }
  }
}
```

§§

5 Error Responses and Messages

This section describes all error codes and messages returned by the Intel VROC Redfish API.

5.1 Error Response

Whenever the server cannot provide all or part of the requested data, it shall respond with the proper HTTP status code indicating an error. Additionally, each error response shall contain JSON message with Redfish Error object (v1.0.1).

The schema URI for the error object is https://www.intel.com/VROC-redfish-error_v1_0_1.

Table 5-1. Redfish Error Object Properties

Error response message properties (Object)		
error	object	The properties that describe an error from a Redfish Service.
{}.code	string	A string indicating a specific MessageId from a Message Registry.
{}.message	string	A human-readable error message corresponding to the message in a Message Registry.
{}.@Message.ExtendedInfo	array	An array of messages describing one or more error messages.
{}.[].MessageArgs	array	This array of message arguments that are substituted for the arguments in the message when looked up in the message registry.
{}.[].MessageId	string	This is the key for this message which can be used to look up the message in a message registry.
{}.[].MessageSeverity	string	This property shall contain the severity of the message. Services can replace the value defined in the message registry with a value more applicable to the implementation. Valid values are *Ok, Warning, Critical*.
{}.[].Resolution	string	Used to provide suggestions on how to resolve the situation that caused the error.

Error response example:

```
{
  "error": {
    "code": "Base.1.8.PropertyValueNotInList",
    "message": "The value 'Example' for the property WriteCachePolicy is not in the list
of acceptable values.",
    "@Message.ExtendedInfo": [{
        "@odata.type": "#Message.v1_1_1.Message",
        "MessageId": "Base.1.8.PropertyValueNotInList",
        "MessageArgs": ["Example", "WriteCachePolicy"],
        "MessageSeverity": "Warning",
        "Resolution": "Choose a value from the enumeration list that the implementation
can support and
```

```
                    resubmit the request if the operation failed."
    }]
  }
}
```

5.2 Supported Errors

The Intel VROC Redfish API supports several types of error messages. All the supported errors are defined in the Redfish Base Registry v.1.8.2 (see section 1.4 References). The below table summarizes all the supported error messages, their severity and short description of when service can return them. For detailed information on error messages, message arguments and suggested resolution, see official registry file. In some cases, the default resolution can be replaced by the service with more detailed information on occurred error.

Table 5-2. Errors Supported by VROC Redfish API

MessageId	Severity	Description
General Errors		
InternalError	Critical	Indicates that the request failed for an unknown internal error but that the service is still operational.
EmptyJSON	Warning	Indicates that the request body contained an empty JSON object when one or more properties are expected in the body.
PropertyMissing	Warning	Indicates that a required property was not supplied as part of the request.
PropertyNotWritable	Warning	Indicates that a property was given a value in the request body, but the property is a read-only property.
PropertyValueConflict	Warning	Indicates that the requested write of a property value could not be completed, because of a conflict with another property value. Example includes a request with Volume resource property **JournalingMedia** set, while **WriteHoleProtectionPolicy** property is unset or with value different than *Journaling*.
PropertyValueIncorrect	Warning	Indicates that the requested write of a property could not be completed, because of an incorrect value of the property such as when it does not meet the constraints of the implementation or match the regular expression requirements. This error covers several situations when submitted request data does not meet the Intel VROC service requirements or is not supported. Detailed information on error shall be included in Resolution property.
ResourceNotfound	Critical	Indicates that the operation expected a resource identifier that corresponds to an existing resource, but one was not found.

MessageId	Severity	Description
General Errors		
ResourceExhaustion	Critical	Indicates that a resource could not satisfy the request due to some unavailability of resources. An example is that available capacity has been allocated.
ResourceInUse	Warning	Indicates that a change was requested to a resource, but the change was rejected due to the resource being in use or transition. Examples include: operations on Volume that is undergoing migration process deleting the first of two volumes in an array (set of drives on which multiple volumes are created), as it would change the UUID of the second volume Detailed information on error shall be included in Resolution property.
POST (Create) Related Errors		
CreateFailedMissingReqProperties	Critical	Indicates that a create was attempted on a resource but that the properties that are required for the create operation were missing from the request.
POST (Action) Related Errors		
ActionNotSupported	Critical	Indicates that the action supplied with the POST operation is not supported by the resource.
ActionParameterMissing	Critical	Indicates that the action requested was missing a parameter that is required to process the action.
ActionParameterValueFormatError	Warning	Indicates that a parameter was given the correct value type, but the value of that parameter was not supported. This includes the value size or length has been exceeded.
ActionParameterNotSupported	Warning	Indicates that the parameter supplied for the action is not supported on the resource.
DELETE Related Errors		
ResourceCannotBeDeleted	Critical	Indicates that a delete operation was attempted on a resource that cannot be deleted. Can occur when attempting to delete a resource that is in a state that it cannot be deleted (ea. System Volume on Windows OS).

§§

6 Technical Details

6.1 Supported Features per OS

The Intel VROC service is a component that acts as a PLDM RDE device. It is a software solution running on a platform host that serves information to BMC. The host must be booted to one of the supported operating systems (Linux*, Windows, UEFI). Some features might not be supported on all OSs. For detailed information on supported features on each operating systems family, see table below.

Table 6-1. Supported Features per OS

VROC Feature	Redfish equivalent	UEFI	Linux**	Windows
Get Drive Inventory	GET on Drive resource	Supported	Supported	Supported
Get Volume Inventory	GET on Volume resource	Supported	Supported	Supported
Get Controller Inventory	GET on Storage Controller resource	Supported	Supported	Supported
Mark Drive as Spare	PATCH on Drive.HotspareType	Supported	Supported	Supported
Unmark Spare Drive	PATCH on Drive.HotspareType	Supported	Supported	Supported
Mark Drive as Normal	PATCH on Drive.StatusIndicator	Supported	Not supported	Supported
Activate Drive LED	PATCH on Drive.LocationIndicatorActive	Supported	Supported	Supported
Create Volume	POST on Volume Collection	Supported	Supported	Supported
Mark Volume as Normal	POST on Volume.ForceEnable (Action)	Supported	Not supported	Supported
Delete Volume	DELETE on Volume	Supported	Supported	Supported
Set RAID Write Hole policy	PATCH on Volume.WriteHoleProtectionPolicy	Supported	Supported	Supported
Initialize Volume	POST on Volume.Initialize (Action)	Not supported	Not supported	Supported
Event notifications	RedfishMessageEvent with Event resource	Supported	Supported	Supported

VROC Feature	Redfish equivalent	UEFI	Linux**	Windows
Set Volatile Write Cache on Drive	PATCH on Drive.WriteCacheEnabled	Not supported	Supported	Supported
Expand Volume	PATCH on Volume.CapacityBytes	Not supported	Supported	Supported
Change RAID Layout	POST on Volume.ChangeRAIDLayout (Action)	Not supported	Supported	Supported
Rename Volume	PATCH on Volume.DisplayName	Not supported	Supported	Supported
Verify Volume	POST on Volume.CheckConsistency (Action)	Not supported	Supported	Supported
Reset To Defaults	POST on Storage.ResetToDefaults (Action)	Supported	Supported	Supported
Sanitize Drive	POST on Drive.SecureErase (Action)	Supported	Supported	Supported
Set Controller Rate	PATCH on StorageController.ControllerRate	Not supported	Supported	Not supported

NOTE: On Windows OS any action that triggers drives enumeration, e.g., unplugging existing drive from the platform, renders VROC service busy and unresponsive for a short time. It may take up few seconds.

6.2 RAID Volume Initialization Methods

RAID volume initialization is synchronizing all redundant data on a volume. Executing the process from the Redfish action will not remove the volume's data. The RAID members' data will be synchronized in case of drive inconsistencies.

There are two ways to invoke RAID volume initialization using VROC Redfish API Initialize Method property.
First is to define Initialize Method property value during RAID volume creation (section 4.3.1), second is to invoke Initialize Method action on already created RAID volume (section 4.2.6).

Below section describes all supported methods of Initialize Method usage by VROC Redfish API.

6.2.1 Supported Values for Initialize Method Property

VROC Redfish API supports two values for Initialize Method property:

- Background: RAID volume data synchronization is performed in the background tasks. Volume is to be used at any time.
- Skip: No RAID volume data synchronization is ongoing. Volume is to be used at any time.

6.2.2 Initialize Method Defined used During RAID Volume Creation

Initialize Method property is an optional request parameter for RAID volume creation operation. RAID 1/5/10 supports “Background” and ”Skip” property values, while RAID 0 supports only “Skip”. For detailed information on all InitializeMethod property usages, see table below.

Table 6-2. Initialize Method Property Values for RAID Volume Creation Pperation.

RAID Volume Level	InitializeMethod property value in request body	InitializeMethod property value after volume creation	Support
RAID 0	Skip	Skip	Supported
RAID 0	Background	-	Not supported
RAID 1/5/10	Skip	Skip	Supported
RAID 1/5/10	Background	Background	Supported
RAID 0	No property defined in request	Skip	Supported
RAID 1/5/10	No property defined in request	Background	Supported

6.2.3 Initialize Method Action to be used on an Already Created RAID Volume

User can invoke an Initialize Method action on the RAID volume, which was created with Initialize Property value “Skip”. RAID volume can be initialized only once, as so Initialize Method action can be performed only once as well.
This action is supported only for Windows OS, and only supported value is ”Background”.

Table 6-3. Initialize Method Action Values

RAID Volume Level	InitializeMethod action value	InitializeMethod property value after action invocation	Support
RAID 0	Background	-	Not supported
RAID 1/5/10	Background	Background	Supported
RAID 0	No property value defined in request	-	Not supported

RAID Volume Level	InitializeMethod action value	InitializeMethod property value after action invocation	Support
RAID 1/5/10	No property value defined in request	Background	Supported

6.2.4 Initialization Progress Verification.

Verification whether initialization process is currently ongoing on the RAID volume can be done by invoking HTTP GET operation on Volume Resource (described in section 3.8)

Response example:

```
{
  "Operations": [{
      "Operation": "Initialize",
      "PercentageComplete": 24
  }],
}
```

For the RAID volume during initialization process, *Operations* object will contain "Operation": "Initialize" with current progress ("PercentageComplete"). Once no "Initializing" operation is reported, RAID volume initialization has been finished or RAID volume was created with Initialize Method property value "Skip".

6.3 Resource Status Mapping

This section describes the mapping of resource statuses from VROC-specific to Redfish properties. There are two resources for which mapping is needed – Volume and Drive.

6.3.1 Volume

The below table shows how VROC-specific Volume state property is mapped to Redfish Status object with State and Health properties.

Table 6-4. Volume State Mapping

[VROC] Volume State	Status.State	Status.Health	Operations.Operation
Normal	Enabled	OK	N/A
Degraded	Enabled	Warning	N/A
Failed	UnavailableOffline	Critical	N/A
Initializing	Enabled	OK	Initialize
Rebuilding	Updating	Warning	Rebuild
Verifying	Enabled	OK	CheckConsistency
Verifying and Repair	Enabled	OK	CheckConsistency
Migrating	Updating	OK	ChangeRAIDType

[VROC] Volume State	Status.State	Status.Health	Operations.Operation
Locked	StandbyOffline	Warning	N/A

NOTE: "Locked" VROC Volume State is supported only on Linux* operating system.

6.3.2 Drive

For Drive resource, the Intel VROC (VMD NVMe RAID) driver collects two properties that affect its status – Drive Status and Drive Usage. Additionally, in case the Drive is part of a Volume resource, the Volume status might affect Drive StatusIndicator. The below tables show how each of VROC-specific properties affect the Redfish properties.

Table 6-5. Drive Status Mapping

[VROC] Drive State	Status.State	Status.Health	StatusIndicator
Offline	StandbyOffline	OK	OK
Failed	UnavailableOffline	Critical	Fail
At risk (SMART event)	Enabled	Warning	PredictiveFailureAnalysis
Incompatible	UnavailableOffline	Critical	Fail
Normal	Enabled	OK	OK
Locked	StandbyOffline	Warning	OK
Unsupported	UnavailableOffline	Critical	Fail

NOTE: "Offline" VROC Drive State is not supported on Linux* operating system.

Table 6-6. Drive Condition Mapping

Condition	MessageSeverity (MessageId)	Status.State	Status.Health	StatusIndicator
Removed	Critical (DriveRemoved)	N/A	N/A	N/A
Inserted	OK (DriveInserted)	Enabled	OK	OK
Failed	Critical (DriveFailure)	Unavaiable Offline	Fail	Fail
Predictive Failure	Critical (DrivePredictiveFailure)	Enabled	Warning	PredictiveFailure Analysis
Predictive Failure Cleared	OK (DrivePredictiveFailureCleared)	Enabled	OK	OK

Table 6-7. Drive Usage Mapping

[VROC] Drive Usage	StatusIndicator	Status.State	Status.Health
Spare Drive	Hotspare	StandbySpare	Ok

In case of multiple Intel VROC properties affect the same Redfish property (StatusIndicator), they are processed with following priority: Drive Status, Volume State, Drive Usage.

Table 6-8. Volume States that affect Drive

[VROC] Volume State	Drive StatusIndicator
Degraded	InACriticalArray
Failed	InAFailedArray
Rebuilding	Rebuild

NOTES: Once Volume State is "Rebuilding" all RAID volume drive's StatusInidicator property will be "Rebuild".

6.4 VROC Software Keys

This section describes impact of VROC Software/Hardware keys on VROC Redfish API functionality.

Scope of functionality provided by the Intel VROC (VMD NVMe RAID) software depends on the Intel VROC Upgrade Keys installed. There are four types of Intel VROC Upgrade keys: *'Intel-SSD-Only','RAID1 Only", 'Standard'* and '*Premium*'. This translates into 5 different configuration SKUs. These different SKUs can be checked under SKU property of Intel VMD Storage Controller resource.

There are two properties directly impacted by the key type. Those are *SKU* and *SupportedRAIDTypes* properties of Intel VMD Storage Controller resource. Below is the table that shows values of this properties for each SKU with additional comment on provided functionality.

Table 6-9. Properties Values on Different SKUs

SKU	SupportedRAIDTypes	Additional Comment
"Pass-Thru"	N/A	Basic drive monitoring. No RAID level is supported.
"Pass-Thru-Trial"	["RAID0", "RAID10", "RAID1", "RAID5"]	There is no Intel VROC Upgrade Keys installed, however a Trial period is activated. Temporarily provides full scope of the Intel VROC Software functionality.
"Intel-SSD-Only"	["RAID0", "RAID10", "RAID1", "RAID5"]	Operations are supported only for Intel NVMe drives.
"Standard"	["RAID0", "RAID10", "RAID1"]	Operations related to RAID5 are not supported

SKU	SupportedRAIDTypes	Additional Comment
		(volume creation, migration etc.).
"Premium"	["RAID0", "RAID10", "RAID1", "RAID5"]	Provides full scope of the Intel VROC Software functionality.
"RAID1 Only"	["RAID1"]	Operations are supported only for RAID1.

6.5 RAID volume migrations

This section describes all supported RAID volume migrations (changing the type of an existing volume on the system) for Windows and Linux* operating systems. RAID volume migration can be invoked using *Change RAID layout* operation.

All supported RAID volume level migrations are shown in table 42. All specific conditions for RAID level are described in table 43.

Table 6-10. Supported RAID Volume Level Migrations

Source Level \ Destination	RAID 0	RAID 1	RAID 10	RAID 5
RAID 0	N/A	No	[1]Yes	Yes
RAID 1	Yes	N/A	No	[2]Yes
RAID 10	[1]Yes	No	N/A	[2]Yes
RAID 5	No	No	No	N/A

[1] Supported only on Linux.**
[2] Supported only on Windows. On Linux* given RAID level migration can be done through indirect migration.

Table 6-11. RAID Level Migrations Specific Conditions

Title	Linux**	Windows
RAID 0 to RAID 5 migraiton	Single drive is required to be added to the RAID 5 when migrating from RAID 0.	At least one drive is required to be added to RAID 5 when migrating from RAID 0.
RAID 0 to RAID 10 migration	Two or multiple of two drivers are required to be added to the RAID 10 when migrating from two-drive RAID 0.	N/A
RAID 1 to RAID 0 migration	No drives are required to be added to the RAID 1 when migrating from RAID 0.	No drives are required to be added to the RAID 1 when migrating from RAID 0.
RAID 1 to RAID 5 migration	No direct migration. Indirect migration: RAID 1 to RAID 0, then RAID0 to RAID 5.	At least one drive is required to be added to the RAID 5 when migrating from RAID 1.

Title	Linux**	Windows
RAID 10 to RAID 0 migration	No drives are required to be added to the RAID 0 when migrating from RAID 10.	N/A
RAID 10 to RAID 5 migration	No direct migration. Indirect migration: RAID 10 to RAID 0, then RAID0 to RAID 5.	No drives are required to be added to the RAID 5 when migrating from RAID 10.
Number of member drives after changing RAID level without providing additional drives	RAID volume migration will be done with the same number of member drives for source RAID volume.	RAID volume migration will be done with the same number of member drives for source RAID volume.

§§

7 PLDM Specifics

As mentioned in the Introduction, the VROC OOB service acts as the **PLDM RDE device.** Redfish Device Enablement (RDE) is only one of a few types defined by the PLDM standard. As such, some of the functionalities required by the PLDM standard, and defined in other types, do not map directly into Redfish API.

This section shall describe, when needed, the details of supported functionality that is not directly accessible via Redfish API.

7.1 Events

The eventing mechanism of PLDM is described in more details in section 16 of *Platform Level Data Model (PLDM) for Platform Monitoring and Control Specification* (see References). From the conditional commands described in that section, VROC OOB service supports:

- GetTerminusUID
- SetEventReceiver – used for general management of events mechanism
- GetEventReceiver
- EventMessageSupported
- EventMessageBufferSize

NOTE: Intel VROC Redfish API supports only synchronous events.

There are two **PLDM Event Types** supported:

- *redfishMessageEvent* – Events compliant with Redfish standard documentation. See section 3.1 for more information on supported events.
- *pldmPDRRepositoryChgEvent* – These events are useful for monitoring any changes to the resources managed by VROC. It is generated when: new resource has been added, existing resource has been removed or Volume resource Status.State field has changed.

7.2 Supported HTTP Operations

Intel VROC OOB service acting as PLDM RDE Device supports following HTTP requests:

- HEAD (all resources)
- GET (all resources)
- POST (selected resources, as described in section 4. Operations)
- PATCH (selected resources, as described in section 4. Operations)
- DELETE (on Volume resource)

§§