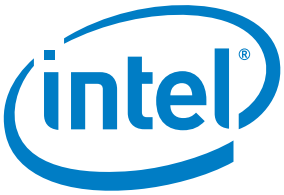


Intel® Rack Scale Design (RSD) Rack Management Module (RMM) Representational State Transfer (REST)

API Specification
Software v2.3.2

September 2018

Revision 003

Document Number: 337205-003US

Contents

1.0 Introduction 6
1.1 Scope 6
1.2 Intended Audience 6
1.3 Conventions 6
1.4 Notes and Symbol Convention 6
1.5 JSON Serialization Convention 7
1.6 HTTP Response Codes 7
1.7 Terminology 7
1.8 Reference Documents 7
2.0 Overview 9
2.1 API Structure and Relation 9
2.2 Rack Management Model and Definitions 10
3.0 RMM REST API Error Codes 11
3.1 API Error Response 11
3.1.1 Message Object 11
3.1.2 Example Error JSON Object 11
3.2 API Error Codes 12
3.2.1 General Error Codes 12
3.2.2 PATCH Method Error Codes 13
4.0 Rack Management Module API Definition 14
4.1 Odata* Support 14
4.2 Asynchronous Operations 14
4.3 Protocol Version 14
4.3.1 Operations 15
4.4 Odata Service Document 15
4.4.1 Operations 15
4.5 Intel® RSD OEM Extensions 16
4.6 Service Root 16
4.6.1 Operations 16
4.7 Manager Collection 17
4.7.1 Operations 17
4.8 Manager 18
4.8.1 Operations 18
4.9 Metric Definition Collection 21
4.9.1 Operations 21
4.10 Metric Definition 22
4.10.1 Operations 22
4.11 TelemetryService 23
4.11.1 Operations 23
4.12 ChassisCollection 24
4.12.1 Operations 25
4.13 Chassis 25
4.13.1 Operations 26
4.14 Power 28
4.14.1 Operations 28
4.15 Thermal 31

4.15.1 Operations 31
4.16 UpdateService 33
4.16.1 Operations 34
4.17 ActionInfo 36
4.17.1 Operations 36
4.18 RMM – PSME Common Resources 37

Figures

Figure 1. Typical Rack Components 10
Figure 2. Chassis Collection Relationship between Components 24
Figure 3. SimpleUpdate Action Component Interactions 35

Tables

Table 1. Terminology 7
Table 2. Reference Documents and Resources 7
Table 3. Resources and Uniform Resource Identifier (URI) 9
Table 4. Rack Management Definitions 10
Table 5 API Error Response Attributes 11
Table 6. API Error Response Attributes 11
Table 7. HTTP Error Status Codes 12
Table 8. Chassis Properties 27
Table 9. Desired Fan Speed Properties 33
Table 10. RMM - PSME Common Resources 37

Revision History

Revision	Description	Date
003US	Intel® RSD minor software v2.3.2 release • Updated Table 3 replacing WIP with V1_0_0 • Added note after Figure 2. Chassis Collection Relationship between Components	September 2018
002US	Minor updates for Intel® RSD software v2.3.1 interim release as part of the RSD v2.3.1 document set	July 2018
001US	Initial release for Intel® RSD Software release v2.3	May 2018

§

1.0 Introduction

1.1 Scope

This document defines the Intel® Rack Scale Design (Intel® RSD) Rack Management Module (RMM) REST API Software v2.3.2.

The interface specified in this document is based on the *Distributed Management Task Force's (DMTF) Redfish* Scalable Platforms API Specification* (DSP0266 1.1.0) and schema (DSP8010 2016.3); refer to Table 2.

1.2 Intended Audience

The intended audience for this document includes designers and engineers working with the Software v2.3.2 release, porting this software to hardware platforms.

1.3 Conventions

The key words/phrases "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119; refer to Table 2.

1.4 Notes and Symbol Convention

Symbol and note convention are similar to typographical conventions used in *Cloud Infrastructure Management Interface 6 (CIMI) Model and REST HTTP-based Protocol 7 an Interface for Managing Cloud Infrastructure*; refer to Table 2.

Notation used in JavaScript Object Notation* (JSON*) serialization description:

- Mandatory in italics indicate data types instead of literal Mandatory
- Characters are appended to items to indicate cardinality:
  - "?" (0 or 1)
  - "*" (0 or more)
  - "+" (1 or more)
- Vertical bars, "|", denote choice. For example, "a|b" means a choice between "a" and "b"
- Parentheses, "(" and ")", are used to indicate the scope of the operators "?", "*", "+" and "|"
- Ellipses (that is, "...") indicate points of extensibility

Note: The lack of ellipses does not mean no extensibility point exists; rather it is just not explicitly called out.

1.5 JSON Serialization Convention

An object is an unordered set of name/value pairs. An object begins with { (left brace) and ends with } (right brace). Each name is followed by: (colon) and the name/value pairs are separated by, (comma).

An array is an ordered collection of values. An array begins with [(left bracket) and ends with] (right bracket). Values are separated by: (comma).

A value can be a string in double quotes; or a number; or true or false or null; or an object or an array. These structures can be nested.

A string is a sequence of zero or more Unicode characters, wrapped in double quotes, using backslash escapes. A character is represented as a single character string. A string is very much like a C or Java* string.

A number is very much like a C or Java number, except that the octal and hexadecimal formats are not used.

1.6 HTTP Response Codes

These are listed in *Scalable Platforms Management API Specification*, Section 6.5.2; refer to Table 2.

1.7 Terminology

Table 1. Terminology

Term	Definition
BMC	Baseboard Management Controller
CIMI	Cloud Infrastructure Management
POD	A physical collection of multiple racks
PODM	POD Manager
Intel® RDC	Intel® Rack Scale Design
JSON*	JavaScript Object Notation*
PSU	Power Supply Unit
RMC	Rack Management Controller
RMM	Rack Management Module
URI	Uniform Resource Identifier
URL	Uniform Resource Locator

1.8 Reference Documents

Table 2. Reference Documents and Resources

Doc ID	Title	Location
337196	*Intel® Rack Scale Design (Intel® RSD) Pooled System Management Engine (PSME) User Guide Software v2.3.2*	*http://www.intel.com/intelRSD*
337197	*Intel® Rack Scale Design (Intel® RSD) Conformance and Software Reference Kit Getting Started Guide Software v2.3.2*	
337198	*Intel® Rack Scale Design (Intel® RSD) POD Manager (PODM) Release Notes Software v2.3.2*	

Doc ID	Title	Location
337199	*Intel® Rack Scale Design (Intel® RSD) POD Manager (PODM) Representational State Transfer (REST) User Guide Software v2.3.2*	
337200	*Intel® Rack Scale Design (Intel® RSD) Pooled System Management Engine (PSME) Release Notes Software v2.3.2*	
337201	*Intel® Rack Scale Design (Intel® RSD) Firmware Extension Specification Software v2.3.2*	
337202	*Intel® Rack Scale Design (Intel® RSD) Storage Services API Specification Software v2.3.2*	
337203	*Intel® Rack Scale Design (Intel® RSD) Architecture Specification Software v2.3.2*	
337204	*Intel® Rack Scale Design (Intel® RSD) POD Manager (PODM) Representational State Transfer (REST) API Specification Software v2.3.2*	
337206	*Intel® Rack Scale Design (Intel® RSD) Generic Assets Management Interface (GAMI) API Software v2.3.2*	
337207	*Intel® Rack Scale Design (Intel® RSD) Pooled System Management Engine (PSME) Representational State Transfer (REST) API Specification Software v2.3.2*	
DSP0263	*Cloud Infrastructure Management Interface 6 (CIMI) Model and REST HTTP-based Protocol 7 An Interface for Managing Cloud Infrastructure*	*https://www.dmtf.org/sites/default/files/standards/documents/DSP0263_1.0.0.pdf*
DSP0266	*Redfish* Scalable Platforms Management API Specification*	*https://dmtf.org/sites/default/files/standards/documents/DSP0266_1.0.0.pdf*
DSP8010	*Redfish SchemaReadMe v2016.3.0*	*https://www.dmtf.org/sites/default/files/standards/documents/DSP8010_2016.3.zip*
RFC2119	*Key Words for Use in RFCs to Indicate Requirement Levels, March 1997*	*https://ietf.org/rfc/rfc2119.txt*
RFC 5789	*IETF PATCH Method for HTTP*	https://tools.ietf.org/html/rfc5789
N/A	*Hypertext Transfer Protocol - HTTP/1.1*	https://dmtf.org/sites/default/files/standards/documents/DSP0266_1.1.0.pdf

§

2.0 Overview

The Intel® RSD RMM REST API v2.3.2 provides the REST-based interface, which allows for full management of the RMM, including asset discovery and configuration.

2.1 API Structure and Relation

Table 3. Resources and Uniform Resource Identifier (URI)

Resource	Schema Version	URI
Service Root	v1_1_1	/redfish/v1
Chassis Collection		/redfish/v1/Chassis Collection
Chassis	V1_2_0	/redfish/v1/Chassis/{chassisID}
Power	V1_1_0	/redfish/v1/Chassis/{chassisID}/Power
Thermal	V1_1_0	/redfish/v1/Chassis/{chassisID}/Thermal
Managers Collection		/redfish/v1/Managers
Managers	V1_2_0	/redfish/v1/Managers/{managerID}
Network Protocol	V1_0_0	/redfish/v1/Managers/{managerID}/NetworkProtocol
Ethernet Interfaces Collection		/redfish/v1/Managers/{managerID}/EthernetInterfaces
Ethernet Interfaces	V1_0_0	/redfish/v1/Managers/{managerID}/EthernetInterfaces/{nicID}
VLAN Network Interfaces Collection		/redfish/v1/Managers/{managerID}/EthernetInterfaces/{nicID}/VLANs
VLANs Network Interfaces	V1_0_0	/redfish/v1/Managers/{managerID}/EthernetInterfaces/{nicID}/VLANs/{vlanID}
EventService	V1_0_0	/redfish/v1/EventService
Event Subscriptions Collection		/redfish/v1/EventService/Subscriptions
Event Subscription	V1_1_0	/redfish/v1/EventService/Subscriptions/{subscriptionID}
TaskService	V1_0_0	/redfish/v1/TaskService
Tasks Collection		/redfish/v1/TaskService/Tasks
Tasks	V1_0_0	/redfish/v1/TaskService/Tasks/{taskID}
TelemetryService	V1_0_0	/redfish/v1/TelemetryService
MetricDefinitions Collection	V1_0_0	/redfish/v1/TelemetryService/MetricDefinitions
MetricDefinitions	V1_0_0	/redfish/v1/TelemetryService/MetricDefinitions/{metricDefinitionId}
UpdateService	V1_0_0	/redfish/v1/UpdateService
ActionInfo	V1_0_0	/redfish/v1/UpdateService/SimpleUpdateActionInfo

2.2 Rack Management Model and Definitions

Figure 1 illustrates typical rack components managed by the Intel® RMM API Specification Software v2.3.2.

Figure 1. Typical Rack Components

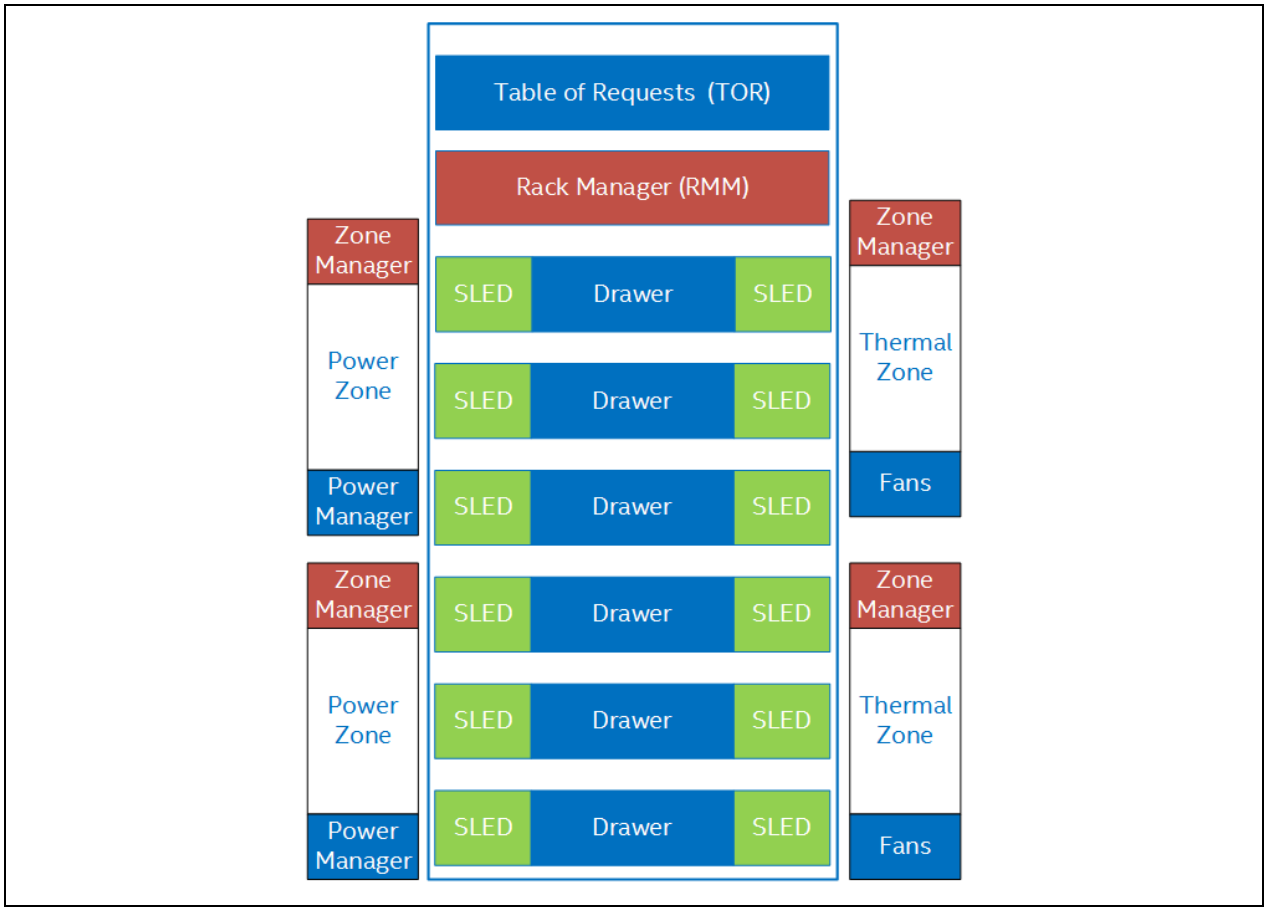


Table 4. Rack Management Definitions

Term	Definition
Rack	Includes one or multiple Power and Thermal Zones.
Power Zone	The Power Zone is one power management domain; the servers in a power zone share the same Power Supply Units (PSUs), including a power shelf (or PSUs) and a number of trays powered by that power shelf.
Thermal Zone	The Thermal Zone is one thermal management domain; the servers in a thermal zone share the same cooling devices (Fans). The devices in the zone cool multiple trays.
Tray/Drawer	Includes one or multiple server modules.
RMM	Rack Management Module. RMM is the rack controller exposing, managing power, and thermal resources. Figure 1 shows the logical concept of the RMM. The rack in Figure 1 contains one RMM.
CM or MBP	Controller Module or Management Backplane. The RMM contains 0 to n CM/MBP.

§

3.0 RMM REST API Error Codes

This chapter provides descriptions of all Error Codes returned by the REST calls implemented in the Intel® RSD RMM REST API of the Intel® RSD software.

3.1 API Error Response

In the case of an error, the PSME REST API responds with an HTTP status code, as defined by the *Hypertext Transfer Protocol - HTTP/1.1* specification (refer to Table 2) and constrained by additional requirements defined in this specification.

HTTP response status codes alone often do not provide enough information to enable deterministic error semantics. The Intel® RSD PSME REST API Software v2.3.2 returns extended error information as a JSON object with a single property named `"error"`. The value of this property is a JSON object with the properties shown in Table 5.

Table 5 API Error Response Attributes

Attribute	Description
`code`	A string indicating a specific `MessageId` from the message registry. `"Base.1.0.GeneralError"` should be used only if there is no better message.
`message`	A human readable error message corresponding to the message in the message registry.
`@Message.ExtendedInfo`	An array of message objects describing one or more error message(s).

3.1.1 Message Object

Message Objects provide additional information about an object, property, or error response. Messages are represented as a JSON object with the following properties:

Table 6. API Error Response Attributes

Attribute	Description
`MessageId`	String indicating a specific error or message (not to be confused with the HTTP status code). This code can be used to access a detailed message from a message registry.
`Message`	A human readable error message indicating the semantics associated with the error. This is the complete message, and does not rely on substitution variables.
`MessageArgs`	An optional array of strings representing the substitution parameter values for the message. This is included in the response if a `MessageId` is specified for a parameterized message.
`Severity`	An optional string representing the severity of the error.
`Resolution`	An optional string describing recommended action(s) to take to resolve the error.
`RelatedProperties`	An optional array of JSON Pointers defining the specific properties within a JSON payload described by the message.

3.1.2 Example Error JSON Object

```
{
    "error": {
        "code": "Base.1.0.GeneralError",
        "message": "A general error has occurred. See ExtendedInfo for more
information.",
        "@Message.ExtendedInfo": [
            {
                "@odata.type" : "/redfish/v1/$metadata#Message.v1_0_0.Message",
```

```
                "MessageId": "Base.1.0.MalformedJSON",
                "Message": "The request body submitted was malformed JSON and could
not be parsed by the receiving service",
                "Severity": "Error"
            }
            {
                "@odata.type" : "/redfish/v1/$metadata#Message.v1_0_0.Message",
                "MessageId": "Base.1.0.PropertyNotWriteable",
                "RelatedProperties": [
                    "#/Name"
                ],
                "Message": "The property Name is a read property and cannot be
assigned a value",
                "MessageArgs": [
                    "Name"
                ],
                "Severity": "Warning",
                "Resolution": "Remove the property from the request body and resubmit
the request if the operation failed"
            }
        ]
    }
}
```

3.2 API Error Codes

In general, if an error isn't described in any of the following tables, it is mapped into an HTTP 500 Internal Error Code.

3.2.1 General Error Codes

For a detailed list of Error Codes, refer to *Redfish* Scalable Platforms Management API Specification*, DSP0266, Section 6.5.2 (refer to Table 2).

Table 7. HTTP Error Status Codes

HTTP Status Code	Description
400 Bad Request	The request is not processed because it contains missing or invalid information (such as a validation error on an input field, a missing required value, and so on). An extended error is returned in the response body.
404 Not Found	The request specified a URI of a resource that does not exist.
405 Method Not Allowed	The HTTP verb specified in the request (for example, DELETE, GET, HEAD, POST, PUT, and PATCH) is not supported for this request URI. The response includes an Allow header, which provides a list of methods supported by the resource identified by the Request-URI.
409 Conflict	A creation or update request could not be completed because it would cause a conflict in the current state of the resources supported by the platform (for example, an attempt to set multiple attributes that work in a linked manner using incompatible values).
500 Internal Server Error	The server encountered an unexpected condition that prevented it from fulfilling the request. An extended error is returned in the response body.
501 Not Implemented	The server does not (currently) support the functionality required to fulfill the request. This is the appropriate response when the server does not recognize the request method and is not capable of supporting it for any resource.
503 Service Unavailable	The server is currently unable to handle the request due to temporary overloading or maintenance of the server.

3.2.2 PATCH Method Error Codes

For the PATCH method, the Intel® RSD service must conform to the *PATCH Method for HTTP RFC 5789* listed in Table 2.

The service will respond with the following error codes in these cases:

- **400 Bad Request** – malformed JSON in the request (values not in range, unknown property, and so on).
- **405 Method Not Allowed** – resource does not support PATCH method.
- **409 Conflict** – update cannot be executed at this moment. User might be able to resolve the conflict and resubmit the request.
- **501 Not Implemented** – resource supports PATCH method, but current implementation does not (for example, underlying HW does not support such functionality).
- **500 Internal Server Error** – all other situations where any of the above codes do not fit (for example, underlying HW does not allow execution of this particular request).

§

4.0 Rack Management Module API Definition

4.1 Odata* Support

Intel® RSD supports the Odata* v4.0 as defined in the *Redfish* Scalable Platforms Management API Specification*; refer to Table 2.

All resources within this Intel® RSD RMM REST API Specification are identified by a unique identifier property named `"@odata.id"`. Resource Identifiers are represented in JSON payloads as Uniform Resource Locator (URL) paths relative to the Redfish Schema portion of the URI. For example, the resource identifiers always start with `"/redfish/"`. The resource identifier is the canonical URL for the resource and can be used to retrieve or edit the resource, as appropriate.

4.2 Asynchronous Operations

While the majority of operations in this architecture are synchronous in nature, some operations take a long time to execute, more time than a client typically wants to wait. For this reason, some operations can be asynchronous at the discretion of the service. The request portion of an asynchronous operation is no different from the request portion of a synchronous operation.

The use of HTTP Response codes enable a client to determine if the operation was completed synchronously or asynchronously. Use of the HTTP Response codes prepares clients to handle both synchronous and asynchronous responses for requests using HTTP DELETE, POST, PATCH and PUT methods.

For details, refer to *Redfish* Scalable Platforms Management API Specification*, Section 8.2, Asynchronous Operations (refer to Table 2).

4.3 Protocol Version

The protocol version is separate from the version of the resources, or the version of the *Redfish* Schema v2016.3*, listed in Table 2, supported by them.

Each version of the Redfish protocol is strongly typed. This is accomplished using the URI of the Redfish service in combination with the resource obtained at that URI, called the `ServiceRoot`.

The root URI for this version of the Redfish protocol is`"/redfish/v1/"`.

While the major version of the protocol is represented in the URI, the major version, minor version, and errata version of the protocol are represented in the version property of the `ServiceRoot` resource, as defined in the Redfish Schema for that resource. The protocol version is a string of the form:

`MajorVersion.MinorVersion.Errata`

Where:

- ***MajorVersion* = integer**: something in the class changed in a backward incompatible way.
- ***MinorVersion* = integer**: a minor update. New functionality may have been added but nothing removed. Compatibility is preserved with previous minor versions.
- ***Errata* = integer**: something in the prior version was broken and needed to be fixed.

Any resource discovered through links found by accessing the root service, or any service or resource referenced using references from the root service, will conform to the same version of the protocol supported by the root service.

4.3.1 Operations

4.3.1.1 GET

Request:

```
GET /redfish
Content-Type: application/json
```

Response:

```
{
        "v1": "/redfish/v1/"
}
```

4.4 Odata Service Document

This Odata Service Document provides a standard format for enumerating the resources exposed by the service, enabling generic hypermedia-driven OData clients to navigate to the resources of the service.

4.4.1 Operations

4.4.1.1 GET

Request:

```
GET /redfish/v1/odata
Content-Type: application/json
```

Response:

```
{
    "@odata.context": "/redfish/v1/$metadata",
    "value": [
        {
            "name": "Service",
            "kind": "Singleton",
            "url": "/redfish/v1/"
        },
        {
            "name": "Chassis",
            "kind": "Singleton",
            "url": "/redfish/v1/Chassis"
        },
        {
            "name": "Managers",
            "kind": "Singleton",
            "url": "/redfish/v1/Managers"
        },
        {
            "name": "EventService",
            "kind": "Singleton",
            "url": "/redfish/v1/EventService"
        },
        {
            "name": "Tasks",
            "kind": "Singleton",
            "url": "/redfish/v1/TaskService"
```

```
        },
        {
            "name": "Registries",
            "kind": "Singleton",
            "url": "/redfish/v1/Registries"
        },
        {
            "name": "UpdateService",
            "kind": "Singleton",
            "url": "/redfish/v1/UpdateService"
        }
    ]
}
```

4.5 Intel® RSD OEM Extensions

All Intel® RSD OEM Extensions to all resources defined in this document are supported.

4.6 Service Root

Service Root resource – entry point.

Properties' details are available in the `ServiceRoot.xml` metadata file.

4.6.1 Operations

4.6.1.1 GET

Request:

```
GET /redfish/v1
Content-Type: application/json
```

Response:

```
{
  "@odata.context": "/redfish/v1/$metadata#ServiceRoot.ServiceRoot",
  "@odata.id": "/redfish/v1/",
  "@odata.type": "#ServiceRoot.v1_1_1.ServiceRoot",
  "Id": "RootService",
  "Name": "Root Service",
  "Description": "description-as-string",
  "RedfishVersion": "1.1.0",
  "UUID": "92384634-2938-2342-8820-489239905423",
  "Chassis": {
    "@odata.id": "/redfish/v1/Chassis"
  },
  "Managers": {
    "@odata.id": "/redfish/v1/Managers"
  },
  "EventService": {
    "@odata.id": "/redfish/v1/EventService"
  },
  "Tasks": {
    "@odata.id": "/redfish/v1/TaskService"
  },
  "TelemetryService": {
```

```
    "@odata.id": "/redfish/v1/TelemetryService"
  },
  "Registries": {
    "@odata.id": "/redfish/v1/Registries"
  },
  "UpdateService": {
    "@odata.id": "/redfish/v1/UpdateService"
  },
  "Oem": {
    "Intel_RackScale": {
      "@odata.type": "#Intel.Oem.ServiceRoot",
      "ApiVersion": "2.2.0",
    }
  },
  "Links": {}
}
```

4.6.1.2 PUT

The PUT operation is not allowed on the service root resource.

4.6.1.3 PATCH

Service Root operation is not allowed on this resource.

4.6.1.4 POST

Service Root operation is not allowed on this resource.

4.6.1.5 DELETE

Service Root operation is not allowed on this resource.

4.7 Manager Collection

The Manager Collection resource provides a collection of all managers available in a rack, manageable through the RMM.

```
Metadata file: ManagerCollection.xml
```

4.7.1 Operations

4.7.1.1 GET

Request:

```
GET /redfish/v1/Managers
Content-Type: application/json
```

Response:

```
{
  "@odata.context": "/redfish/v1/$metadata#ManagerCollection.ManagerCollection",
  "@odata.id": "/redfish/v1/Managers",
  "@odata.type": "#ManagerCollection.ManagerCollection",
  "Name": "Manager Collection",
  "Description": "description-as-string",
```

```
  "Members@odata.count": 2,
  "Members": [
    {
      "@odata.id": "/redfish/v1/Managers/RackManager"
    },
    {
      "@odata.id": "/redfish/v1/Managers/ZoneManager"
    }
  ]
}
```


4.7.1.2 PUT

Manager Collection operation is not allowed on this resource.

4.7.1.3 PATCH

Manager Collection operation is not allowed on this resource.

4.7.1.4 POST

Manager Collection operation is not allowed on this resource.

4.7.1.5 DELETE

Manager Collection operation is not allowed on this resource.

4.8 Manager

The Manager is a systems management entity, which may implement or provide access to a Redfish service. Examples of managers are Baseboard Management Controllers (BMCs), Enclosure Managers, Management Controllers, and other subsystems that assign manageability functions. There can be multiple Managers in an implementation, and they may or may not be directly accessible through a Redfish-defined interface.

Properties' details are available in the `Manager.xml` metadata file.

4.8.1 Operations

4.8.1.1 GET

Request:

```
GET /redfish/v1/Managers/RackManager
Content-Type: application/json
```

Response:


```
{
  "@odata.context": "/redfish/v1/$metadata#Manager.Manager",
  "@odata.id": "/redfish/v1/Managers/RackManager",
  "@odata.type": "#Manager.v1_2_0.Manager",
  "Id": "RackManager",
  "Name": "Manager",
  "ManagerType": "RackManager",
  "Description": "RackScale RMC",
  "ServiceEntryPointUUID": "11384622-2938-2342-8820-489239905423",
```


```
  "UUID": "00000000-0000-0000-0000-000000000000",
  "Model": "Joo Janta 200",
  "DateTime": "2015-03-13T04:14:33+06:00",
  "DateTimeLocalOffset": "+06:00",
  "PowerState": null,
  "Status": {
    "State": "Enabled",
    "Health": "OK",
    "HealthRollup": null
  },
  "GraphicalConsole": {
    "ServiceEnabled": true,
    "MaxConcurrentSessions": 2,
    "ConnectTypesSupported": [
      "KVMIP"
    ]
  },
  "SerialConsole": {
    "ServiceEnabled": true,
    "MaxConcurrentSessions": 1,
    "ConnectTypesSupported": [
      "Telnet",
      "SSH",
      "IPMI"
    ]
  },
  "CommandShell": {
    "ServiceEnabled": true,
    "MaxConcurrentSessions": 4,
    "ConnectTypesSupported": [
      "Telnet",
      "SSH"
    ]
  },
  "FirmwareVersion": "2.1.71.0",
  "NetworkProtocol": {
    "@odata.id": "/redfish/v1/Managers/RackManager1/NetworkProtocol"
  },
  "EthernetInterfaces": {
    "@odata.id": "/redfish/v1/Managers/RackManager1/EthernetInterfaces"
  },
  "Links": {
    "ManagerForServers": [],
    "ManagerForChassis": [{
      "@odata.id": "/redfish/v1/Chassis/Rack1"
    }],
    "ManagerInChassis": {
      "@odata.id": "/redfish/v1/Chassis/Rack1"
    },
    "ManagerForSwitches": [],
    "Oem": {}
  },
  "Oem": {},
  "PowerState": "On",
  "Actions": {
    "#Manager.Reset": {
      "target": "/redfish/v1/Managers/RackManager/Actions/Manager.Reset",
      "ResetType@Redfish.AllowableValues": ["GracefulRestart"]
    },
    "Oem": {
```

```
      "#Intel_RackScale.LoadFactoryDefaults": {
        "target":
"/redfish/v1/Managers/RackManager/Actions/Oem/Intel_RackScale.LoadFactoryDefaults"
      }
    }
  }
}
```

4.8.1.2 PUT

The manager operation is not allowed on this resource.

4.8.1.3 PATCH

The manager operation is not allowed on this resource.

4.8.1.4 POST

The manager operation is not allowed on this resource.

4.8.1.4.1 Manager Reset

Manager Reset can be initiated using the action below.

Request:

```
POST /redfish/v1/Managers/RackManager/Actions/Manager.Reset
Content-Type: application/json

{
        "ResetType": "GracefulRestart"
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or (when task is created)

```
HTTP/1.1 202 Accepted
Location: http://<ip>:<port>/redfish/v1/TaskService/TaskMonitors/1
{
    "@odata.context": "/redfish/v1/$metadata#Task.Task",
    "@odata.id": "/redfish/v1/TaskService/Tasks/1",
    "@odata.type": "#Task.v1_0_0.Task",
    "Id": "1",
    "Name": "Task 1",
    "TaskState": " New",
    "StartTime": "2016-09-01T04:45+01:00",
    "TaskStatus": "OK",
    "Messages": [
    ]
}
```

4.8.1.4.2 Reset to Factory Defaults

The Rack manager may support a Reset to Factory Defaults. The following request action performs such a reset.

Request:

```
POST /redfish/v1/Managers/RackManager/Actions/Oem/Intel_RackScale.LoadFactoryDefault
```

```
Content-Type: application/json
{
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or (when task is created)

```
HTTP/1.1 202 Accepted
Location: http://<ip>:<port>/redfish/v1/TaskService/TaskMonitors/1
{
    "@odata.context": "/redfish/v1/$metadata#Task.Task",
    "@odata.id": "/redfish/v1/TaskService/Tasks/1",
    "@odata.type": "#Task.v1_0_0.Task",
    "Id": "1",
    "Name": "Task 1",
    "TaskState": "New",
    "StartTime": "2016-09-01T04:45+01:00",
    "TaskStatus": "OK",
    "Messages": [
    ]
}
```

4.8.1.5 DELETE

Reset to factory defaults operation is not allowed on this resource.

4.9 Metric Definition Collection

Property details are available in `MetricDefinitionCollection.xml` metadata file.

4.9.1 Operations

4.9.1.1 GET

Request:

```
GET /redfish/v1/TelemetryService/MetricDefinitions
Content-Type: application/json
```

Response:

```
{
  "@odata.context":
"/redfish/v1/$metadata#TelemetryService/MetricDefinitions/$entity",
  "@odata.id": "/redfish/v1/TelemetryService/MetricDefinitions",
  "@odata.type": "#MetricDefinitionCollection.MetricDefinitionCollection",
  "Name": "Metric Definitions Collection",
  "Description": "description-as-string",
  "Members@odata.count": 2,
  "Members": [
    {
      "@odata.id": "/redfish/v1/TelemetryService/MetricDefinitions/FanSpeedRPM"
    },
    {
      "@odata.id": "/redfish/v1/TelemetryService/MetricDefinitions/RackTemperature"
```

```
    }
  ]
}
```

4.9.1.2 PUT

Metric definition collection operation is not allowed on this resource.

4.9.1.3 PATCH

Metric definition collection operation is not allowed on this resource.

4.9.1.4 POST

Metric definition collection operation is not allowed on this resource.

4.9.1.5 DELETE

Metric definition collection operation is not allowed on this resource.

4.10 Metric Definition

Property details are available in the `MetricDefinition.xml` metadata file. `MetricDefinition` describes either metric associated with physical sensor (for instance, exposed by BMC) or metric associated with specific resource (for instance, statistics of Rack Power Module). This resource is optional for metrics and required for sensors.

4.10.1 Operations

4.10.1.1 GET

Request:

```
GET /redfish/v1/TelemetryService/MetricDefinitions/RackTemperature
Content-Type: application/json
```

Response:

```
{
  "@odata.context": "/redfish/v1/$metadata#MetricDefinition.MetricDefinition",
  "@odata.id": "/redfish/v1/TelemetryService/MetricDefinitions/SLEDTemperatures",
  "@odata.type": "#MetricDefinition.v1_0_0.MetricDefinition",
  "Description": "Zone PSU Temperature MetricDefinition",
  "Name": "Zone Power Supply Unit Temperature definition",
  "Id": "RackTemperature",
  "SensorType": "Temperature",
  "Implementation": "Physical",
  "SensingInterval": "PT1S",
  "MetricType": "Numeric",
  "PhysicalContext": "Backplane",
  "Units": "Cel",
  "MinReadingRange": 0,
  "MaxReadingRange": 110,
  "Precision": 1,

  "MetricProperties": [
    "/redfish/v1/Chassis/Zone1/Thermal#/Temperatures/0/ReadingCelsius"
```

```
  ]
}
```

4.10.1.2 PUT

Metric definition operation is not allowed on this resource.

4.10.1.3 PATCH

Metric definition operation is not allowed on this resource.

4.10.1.4 POST

Metric definition operation is not allowed on this resource.

4.10.1.5 DELETE

Metric definition operation is not allowed on this resource.

4.11 TelemetryService

Property details are available in the `TelemetryService.xml` metadata file.

4.11.1 Operations

4.11.1.1 GET

Request:

```
GET /redfish/v1/TelemetryService
Content-Type: application/json
```

Response:

```
{
    "@odata.context": "/redfish/v1/$metadata#TelemetryService",
    "@odata.type": "#TelemetryService.v1_0_0.TelemetryService",
    "@odata.id": "/redfish/v1/TelemetryService",
    "Id": "TelemetryService",
    "Name": "Telemetry Service",
    "Status": {
        "State": "Enabled",
        "Health": "OK"
    },

    "MetricDefinitions": {
        "@odata.id": "/redfish/v1/TelemetryService/MetricDefinitions"
    }
}
```

4.11.1.2 PUT

The `TelemetryService` operation is not allowed on this resource.

4.11.1.3 PATCH

The `TelemetryService` operation is not allowed on this resource.

4.11.1.4 POST

The `TelemetryService` operation is not allowed on this resource.

4.11.1.5 DELETE

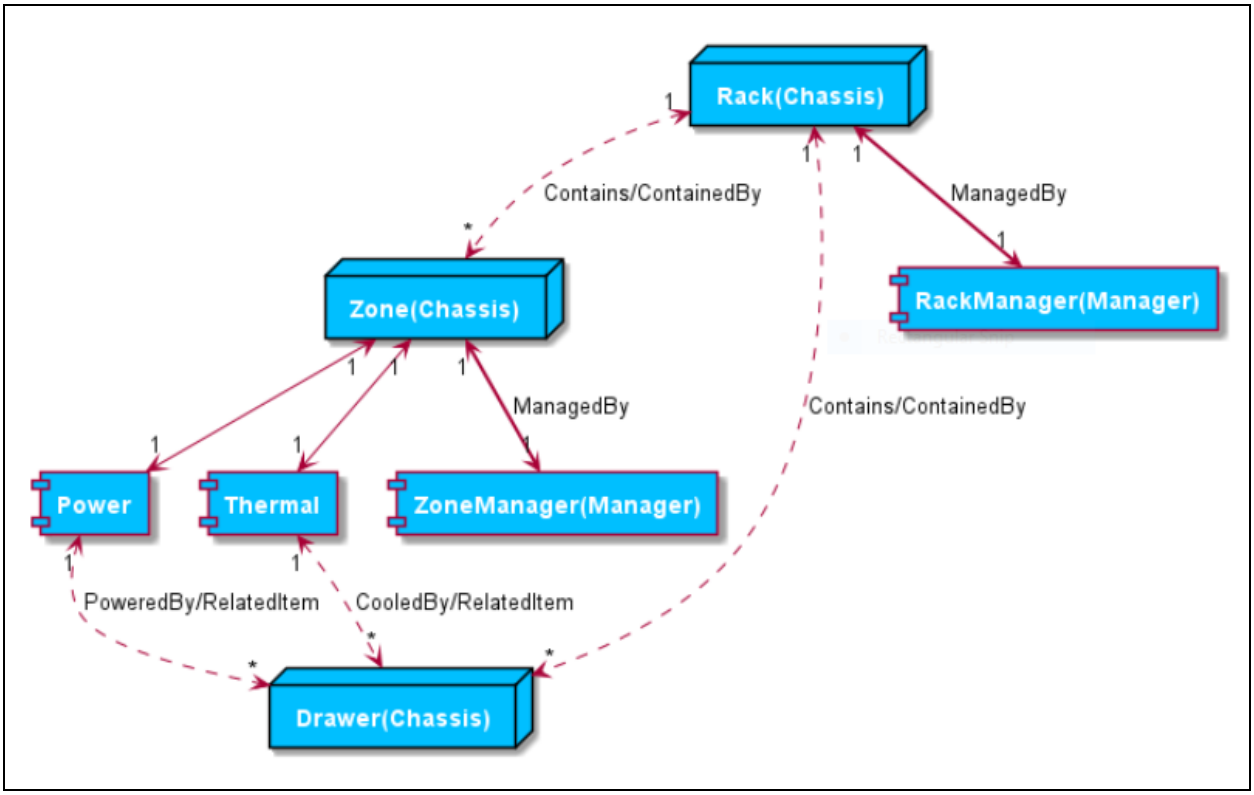
The `TelemetryService` operation is not allowed on this resource.

4.12 ChassisCollection

The `ChassisCollection` resource shown in Figure 2 illustrates the relationship between various chassis components in the Intel® RSD Rack.

Property details are available in the `ChassisCollection.xml` metadata file.

Figure 2. Chassis Collection Relationship between Components



Note: ***The numbers represent a one-to-one or to-many relationship between components, "*" denotes zero or more relationship.***

4.12.1 Operations

4.12.1.1 GET

Request:

```
GET /redfish/v1/Chassis
Content-Type: application/json
```

Response:

```
{
    "@odata.context": "/redfish/v1/$metadata#Chassis",
    "@odata.id": "/redfish/v1/Chassis",
    "@odata.type": "#ChassisCollection.ChassisCollection",
    "Name": "Chassis Collection",
    "Members@odata.count": 3,
    "Members": [
        {
            "@odata.id": "/redfish/v1/Chassis/Rack1"
        },
        {
            "@odata.id": "/redfish/v1/Chassis/Zone1"
        },
        {
            "@odata.id": "/redfish/v1/Chassis/Drawer1"
        }
    ]
}
```

4.12.1.2 PUT

The `ChassisCollection` operation is not allowed on this resource.

4.12.1.3 PATCH

The `ChassisCollection` operation is not allowed on this resource.

4.12.1.4 POST

The `ChassisCollection` operation is not allowed on this resource.

4.12.1.5 DELETE

The `ChassisCollection` operation is not allowed on this resource.

4.13 Chassis

This is the schema definition for the Chassis resource, which represents the properties of the physical components for any system. This one resource is intended to represent racks, rackmount servers, blades, modular systems, enclosures, and all other containers. The non-CPU/device centric parts of the schema are all accessed either directly or indirectly through this resource.

Details of this resource are described in metadata file: `Chassis.xml`

4.13.1 Operations

4.13.1.1 GET

Request:

```
GET /redfish/v1/Chassis/Rack1
Content-Type: application/json
```

Response:


```
{
  "@odata.context": "/redfish/v1/$metadata#Chassis/Members/$entity",
  "@odata.id": "/redfish/v1/Chassis/Rack1",
  "@odata.type": "#Chassis.v1_2_0.Chassis",
  "Id": "Rack1",
  "ChassisType": "Rack",
  "Name": "name-as-string",
  "Description": "description-as-string",
  "Manufacturer": "Intel Corporation",
  "Model": "RackScale_Rack",
  "SKU": "sku-as-string",
  "SerialNumber": "serial-number-as-string",
  "PartNumber": "part-number-as-string",
  "AssetTag": null,
  "IndicatorLED": null,
  "PowerState": null,
  "Status": {
    "State": "Enabled",
    "Health": "OK",
    "HealthRollup": null
  },
  "Oem": {
    "Intel_RackScale": {
      "@odata.type": "Intel.Oem.RackChassis",
      "Location": {
        "Id": "Rack1",
        "ParentId": null,
      },
      "RackSupportsDisaggregatedPowerCooling": false,
      "UUID": "123-124-134-234-13423534",
      "GeoTag": "1.234234, 54.234234"
    }
  },
  "Links": {
    "@odata.type": "#Chassis.v1_2_0.Links",
    "Contains": [{
      "@odata.id": "/redfish/v1/Chassis/Drawer1"
    }, {
      "@odata.id": "/redfish/v1/Chassis/Zone1"
    }
      }],
    "ContainedBy": [],
    "ComputerSystems": [],
    "ManagedBy": {
      "@odata.id": "/redfish/v1/Managers/RackManager1"
    },
    "ManagersInChassis": [{
      "@odata.id": "/redfish/v1/Managers/RackManager1"
    }],
    "PoweredBy": [],
```

```
    "CooledBy": [],
    "Storage": [],
    "Drives": [],
    "Oem": {
      "Intel_RackScale": {
        "@odata.type": "#Intel.Oem.ChassisLinks",
        "Switches": []
      }
    }
  },
  "PowerState": "On",
    "Thermal": {
        "@odata.id": "/redfish/v1/Chassis/Rack1/Thermal"
    },
    "Power": {
        "@odata.id": "/redfish/v1/Chassis/Rack1/Power"
    },
  "Actions": {
    "#Chassis.Reset": {
      "target": "/redfish/v1/Chassis/Rack1/Actions/Chassis.Reset",
      "ResetType@Redfish.AllowableValues": [
      ]
    }
  }
}
```

4.13.1.2 PUT

Chassis operation is not allowed on this resource.

4.13.1.3 PATCH

The PATCH operation can update the properties listed in Table 8.

Table 8. Chassis Properties

Attribute	Type	Required	Description
`AssetTag`	String	No	The user assigned asset tag for this chassis.
`Oem->Intel_RackScale->Location`	Object	No	Object representing the physical location of the chassis. Valid only for resource type `"Rack"`. Following properties can be patched: `"Id"` - String containing physical location ID of this chassis.
`Oem->Intel_RackScale->GeoTag`	String	No	GeoTag – only for Rack chassis.

Request:

```
PATCH /redfish/v1/Chassis/1
Content-Type: application/json
{
    "AssetTag": "My rack"
    "Oem": {
        "Intel_RackScale": {
          "Location": {
            "Id": "Rack_1"
          }
        }
    }
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or:

```
HTTP/1.1 200 OK
{
(updated resource body)
}
```

4.13.1.4 POST

`Chassis.reset` can be initiated using the action below:

Request:

```
POST /redfish/v1/Chassis/Drawer1/Actions/Chassis.Reset
Content-Type: application/json
{
         "ResetType": "ForceRestart"
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or (when task is created)

```
HTTP/1.1 202 Accepted
Location: http://<ip>:<port>/redfish/v1/TaskService/TaskMonitors/1
{
    "@odata.context": "/redfish/v1/$metadata#Task.Task",
    "@odata.id": "/redfish/v1/TaskService/Tasks/1",
    "@odata.type": "#Task.v1_0_0.Task",
    "Id": "1",
    "Name": "Task 1",
    "TaskState": " New",
    "StartTime": "2016-09-01T04:45+01:00",
    "TaskStatus": "OK",
    "Messages": [
    ]
}
```

4.13.1.5 DELETE

The chassis operation is not allowed on this resource.

4.14 Power

Power metrics resource represents the properties for Power Consumption and Power Limiting.

Detailed information about this property can be obtained from metadata file: `Power.xml`

4.14.1 Operations

4.14.1.1 GET

Request:

```
GET /redfish/v1/Chassis/Zone1/Power
Content-Type: application/json
```

Response:


```
{
  "@odata.context": "/redfish/v1/$metadata#Power.Power",
  "@odata.id": "/redfish/v1/Chassis/Zone1/Power",
  "@odata.type": "#Power.v1_1_0.Power",
  "Id": "Power",
  "Name": "Power",
  "Description": "Power",
  "PowerControl": [ {
    "@odata.id": "/redfish/v1/Chassis/Zone1/Power#/PowerControl/0",
    "MemberId": "0",
    "Name": "System Power Control",
    "PowerConsumedWatts": 8000,
    "PowerRequestedWatts": 8500,
    "PowerAvailableWatts": 8500,
    "PowerCapacityWatts": 10000,
    "PowerAllocatedWatts": 8500,
    "PowerMetrics": {
      "IntervalInMin": null,
      "MinConsumedWatts": null,
      "MaxConsumedWatts": null,
      "AverageConsumedWatts": null
    },
    "PowerLimit": {
      "LimitInWatts": null,
      "LimitException": null,
      "CorrectionInMs": null
    },
    "RelatedItem": [ {
      "@odata.id": "/redfish/v1/Chassis/Drawer1"
    } ],
    "Status": {
      "State": "Enabled",
      "Health": "OK",
      "HealthRollup": "OK"
    },
    "Oem": {
    }
  } ],
  "Voltages": [ {
    "@odata.id": "/redfish/v1/Chassis/Zone1/Power#/Voltages/0",
    "MemberId": "0",
    "Name": "VRM1 Voltage",
    "SensorNumber": 11,
    "Status": {
      "State": "Enabled",
      "Health": "OK"
    },
    "ReadingVolts": 12,
    "UpperThresholdNonCritical": null,
    "UpperThresholdCritical": null,
    "UpperThresholdFatal": null,
    "LowerThresholdNonCritical": null,
```

```
    "LowerThresholdCritical": null,
    "LowerThresholdFatal": null,
    "MinReadingRange": null,
    "MaxReadingRange": null,
    "PhysicalContext": "VoltageRegulator",
    "RelatedItem": [ {
      "@odata.id": "/redfish/v1/Chassis/Drawer1"
    } ]
  } ],
  "PowerSupplies": [ {
    "@odata.id": "/redfish/v1/Chassis/Zone1/Power#/PowerSupplies/0",
    "MemberId": "0",
    "Name": "Power Supply Bay 1",
    "Status": {
      "State": "Enabled",
      "Health": "Warning"
    },
    "Oem": {
    },
    "PowerSupplyType": "DC",
    "LineInputVoltageType": "DCNeg48V",
    "LineInputVoltage": -48,
    "PowerCapacityWatts": 400,
    "LastPowerOutputWatts": 192,
    "Model": "499253-B21",
    "Manufacturer": "ManufacturerName",
    "FirmwareVersion": "1.00",
    "SerialNumber": "1z0000001",
    "PartNumber": "1z0000001A3a",
    "SparePartNumber": null,
    "InputRanges": [],
    "IndicatorLED": "Off",
    "RelatedItem": [ {
      "@odata.id": "/redfish/v1/Chassis/Drawer1"
    } ]
  } ],
  "Oem": {
    "Intel_RackScale": {
      "@odata.type": "#Intel.Oem.Power",
      "Actions": {
        "#Intel.Oem.RequestPowerSupplyStateChange": {
          "target":
"/redfish/v1/Chassis/Zone1/Power/Oem/Intel_RackScale/Actions/Intel.Oem.RequestPowerSup
plyStateChange",
          "State@AllowableValues": ["Enabled", "Disabled"],
          "MemberId@AllowableValues": ["0"]
        }
      }
    }
  }
}
```


4.14.1.2 PUT

The power operation is not allowed on this resource.

4.14.1.3 PATCH

The power operation is not allowed on this resource.

4.14.1.4 POST

Power supplies can be enabled/disabled using the following action:

Request:

```
POST
/redfish/v1/Chassis/Zone1/Power/Oem/Intel_RackScale/Actions/Intel.Oem.RequestPowerSupp
lyStateChange
Content-Type: application/json
{
        "State": "Disabled",
        "MemberId”: "0"
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or (when task is created)

```
HTTP/1.1 202 Accepted
Location: http://<ip>:<port>/redfish/v1/TaskService/TaskMonitors/1
{
    "@odata.context": "/redfish/v1/$metadata#Task.Task",
    "@odata.id": "/redfish/v1/TaskService/Tasks/1",
    "@odata.type": "#Task.v1_0_0.Task",
    "Id": "1",
    "Name": "Task 1",
    "TaskState": " New",
    "StartTime": "2016-09-01T04:45+01:00",
    "TaskStatus": "OK",
    "Messages": [
    ]
}
```

4.14.1.5 DELETE

The power operation is not allowed on this resource.

4.15 Thermal

Thermal metrics resource represents the properties for Temperature and Cooling.

Detailed information about the resource's properties can be obtained from the metadata file: `Thermal.xml`

4.15.1 Operations

4.15.1.1 GET

Request:

```
GET /redfish/v1/Chassis/Zone1/Thermal
Content-Type: application/json
```

Response:

```
{
  "@odata.context": "/redfish/v1/$metadata#Thermal.Thermal",
  "@odata.id": "/redfish/v1/Chassis/Zone1/Thermal",
```

```
  "@odata.type": "#Thermal.v1_1_0.Thermal",
  "Id": "Thermal",
  "Name": "Thermal",
  "Description": "Thermal",
  "Temperatures": [ {
    "@odata.id": "/redfish/v1/Chassis/Zone1/Thermal#/Temperatures/0",
    "MemberId": "0",
    "Name": "Drawer inlet Temp",
    "SensorNumber": 42,
    "Status": {
      "State": "Enabled",
      "Health": "OK"
    },
    "ReadingCelsius": 21,
    "UpperThresholdNonCritical": null,
    "UpperThresholdCritical": null,
    "UpperThresholdFatal": null,
    "LowerThresholdNonCritical": null,
    "LowerThresholdCritical": null,
    "LowerThresholdFatal": null,
    "MinReadingRangeTemp": null,
    "MaxReadingRangeTemp": null,
    "PhysicalContext": "Intake",
    "RelatedItem": [ {
      "@odata.id": "/redfish/v1/Chassis/Drawer1"
    } ]
  } ],
  "Fans": [ {
    "@odata.id": "/redfish/v1/Chassis/Zone1/Thermal#/Fans/0",
    "MemberId": "0",
    "Name": "BaseBoard System Fan",
    "PhysicalContext": "Backplane",
    "Status": {
      "State": "Enabled",
      "Health": "OK"
    },
    "Reading": 2100,
    "ReadingUnits": "RPM",
    "UpperThresholdNonCritical": null,
    "UpperThresholdCritical": null,
    "UpperThresholdFatal": null,
    "LowerThresholdNonCritical": null,
    "LowerThresholdCritical": null,
    "LowerThresholdFatal": null,
    "MinReadingRange": null,
    "MaxReadingRange": null,
    "RelatedItem": [ {
      "@odata.id": "/redfish/v1/Chassis/Drawer1"
    } ]
  } ],
  "Oem": {
    "Intel_RackScale": {
      "@odata.type": "#Intel.Oem.Thermal",
      "VolumetricAirflowCfm": 100,
      "DesiredSpeedRpm": 3000,
      "DesiredSpeedPwm": 50
    }
  }
}
```

4.15.1.2 PUT

Thermal operation is not allowed on this resource.

4.15.1.3 PATCH

The following properties can be updated by the PATCH operation:

Table 9. Desired Fan Speed Properties

Attribute	Type	Required	Description
`Oem->Intel_RackScale->DesiredSpeedPwm`	Number	No	This property represents the desired speed of all FANs in the current chassis as a percentage of maximum fan speed. Allowed values are in range from 0 to 100 percent.

Request:

```
PATCH /redfish/v1/Chassis/1
Content-Type: application/json
{
    "AssetTag": "My rack"
    "Oem": {
         "Intel_RackScale": {
          "DesiredSpeedPwm": 90
         }
    }
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or:

```
HTTP/1.1 200 OK
{
(updated resource body)
}
```

4.15.1.4 POST

The PATCH operation is not allowed on this resource.

4.15.1.5 DELETE

The PATCH operation is not allowed on this resource.

4.16 UpdateService

`UpdateService` resource represents the properties required to invoke the software/firmware update.

Note: In the current release, only the Manager Resources can be updated.

4.16.1 Operations

4.16.1.1 GET

Request:

```
GET /redfish/v1/UpdateService
Content-Type: application/json
```

Response:

```
{
  "@odata.type": "#UpdateService.v1_0_2.UpdateService",
  "Id": "UpdateService",
  "Name": "Update service",
  "Status": {
    "State": "Enabled",
    "Health": "OK",
    "HealthRollup": "OK"
  },
  "ServiceEnabled": true,
  "Actions": {
    "#UpdateService.SimpleUpdate": {
      "target": "/redfish/v1/UpdateService/Actions/SimpleUpdate",
      "@Redfish.ActionInfo": "/redfish/v1/UpdateService/SimpleUpdateActionInfo"
    },
    "Oem": {}
  },
  "Oem": {},
  "@odata.context": "/redfish/v1/$metadata#UpdateService/$entity",
}
```

4.16.1.2 PUT

The `UpdateService` operation is not allowed on this resource.

4.16.1.3 PATCH

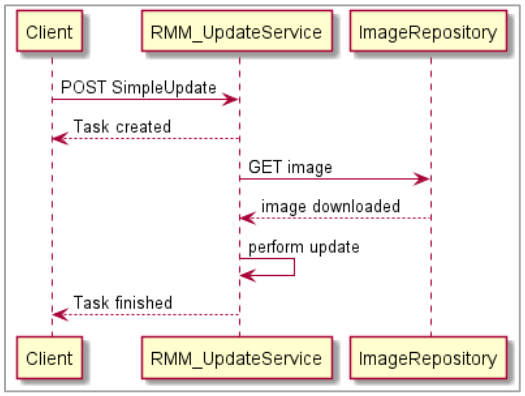
The `UpdateService` operation is not allowed on this resource.

4.16.1.4 POST

4.16.1.4.1 SimpleUpdate Action

The software/firmware update can be initiated using `SimpleUpdate` action. Figure 3 illustrates the interaction between components.

Figure 3. SimpleUpdate Action Component Interactions



Request:

```
POST /redfish/v1/UpdateService/Actions/SimpleUpdate
Content-Type: application/json

{
        "ImageURI": "http://10.0.0.1/images/rmm_image.deb",
        "Targets": [
          "/redfish/v1/Managers/RackManager"
        ],
        "TransferProtocol": "HTTP"
}
```

Response:

```
HTTP/1.1 204 No Content
```

Or (when task is created)

```
HTTP/1.1 202 Accepted
Location: http://<ip>:<port>/redfish/v1/TaskService/TaskMonitors/1
{
    "@odata.context": "/redfish/v1/$metadata#Task.Task",
    "@odata.id": "/redfish/v1/TaskService/Tasks/1",
    "@odata.type": "#Task.v1_0_0.Task",
    "Id": "1",
    "Name": "Task 1",
    "TaskState": " New",
    "StartTime": "2016-09-01T04:45+01:00",
    "TaskStatus": "OK",
    "Messages": [
    ]
}
```

4.16.1.5 DELETE

The `SimpleUpdate` action operation is not allowed on this resource.

4.17 ActionInfo

`ActionInfo` describes the parameters and other information necessary to perform a Redfish Action to a particular action target. As parameter support may differ between implementations and even among instances of a resource, this data can be used to ensure action requests from applications contain supported parameters.

4.17.1 Operations

4.17.1.1 GET (UpdateService/SimpleUpdateActionInfo)

Request:

```
GET /redfish/v1/UpdateService/SimpleUpdateActionInfo
Content-Type: application/json
```

Response:

```
{
  "@odata.type": "#ActionInfo.v1_0_0.ActionInfo",
  "Parameters": [
    {
      "Name": "ImageURI",
      "Required": true,
      "DataType": "String"
    },
    {
      "Name": "TransferProtocol",
      "Required": false,
      "DataType": "String",
      "AllowableValues": [ "HTTP", "HTTPS", "FTP" ]
    },
    {
      "Name": "Targets",
      "Required": false,
      "DataType": "StringArray",
      "AllowableValues": ["RackManager", "ZoneManager"]
    }
  ],
  "Oem": {},
  "@odata.context": "/redfish/v1/$metadata#ActionInfo.ActionInfo",
  "@odata.id": "/redfish/v1/UpdateService/SimpleUpdateActionInfo"
}
```

4.17.1.2 PUT

The `UpdateService/SimpleUpdateActionInfo` operation is not allowed on this resource.

4.17.1.3 PATCH

The PATCH operation is not allowed on `UpdateService/SimpleUpdateActionInfo` resource.

4.17.1.4 POST

The `UpdateService/SimpleUpdateActionInfo` operation is not allowed on this resource.

4.17.1.5 DELETE

The `UpdateService/SimpleUpdateActionInfo` operation is not allowed on this resource.

4.18 RMM – PSME Common Resources

Resources mentioned in Table 10 are shared in the Intel® RSD PSME REST API and Intel® RSD RMM REST API Specifications as common resources. Refer *Intel® RSD PSME REST API Specification* for resource definition, Table 33, Required Resources per Service Type (refer to Table 2).

Table 10. RMM - PSME Common Resources

Resource Name	Supported Operations				
	GET	PATCH	POST	DELETE	Actions
`EventService`	X				
`EventSubscription`	X		X	X	
`MetricDefinition`	X				
`MetricDefinitionCollection`	X				
`MessageRegistryFile`	X				
`Ethernet Interfaces`	X				
`Network Protocol`	X				
`Registries`	X				
`Task`	X				
`TaskCollection`	X				
`TaskService`	X				
`TelemetryService`	X				
`VLAN`	X				

§