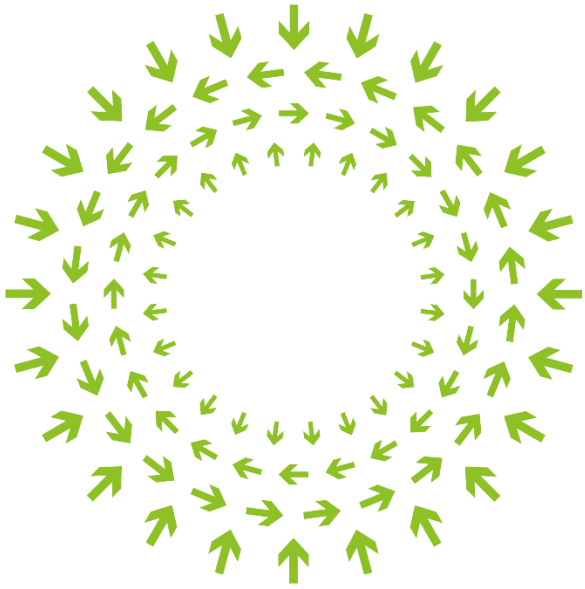

OPEN

Compute Project

OpenRMC Design Specification v1.0.0

July 15, 2020

Table of Contents

1. LICENSE

Contributions to this Specification are made under the terms and conditions set forth in a modified Open Web Foundation Contributor License Agreement ("OWF CLA 1.0") ("Contribution License"). This modified OWF CLA 1.0 can be made available upon request.

The contributions are made by:

Inspur

Intel Corporation

Wiwynn

2. SCOPE

This document specifies the northbound interface of the OpenRMC rack management controller and how to contribute the source code which implements the firmware that executes on the rack manager contribution.

3. OVERVIEW

Figure 1 shows the functional architecture of an OpenRMC rack management controller or rack manager. The rack manager provides a rack manager service by executing firmware or software.

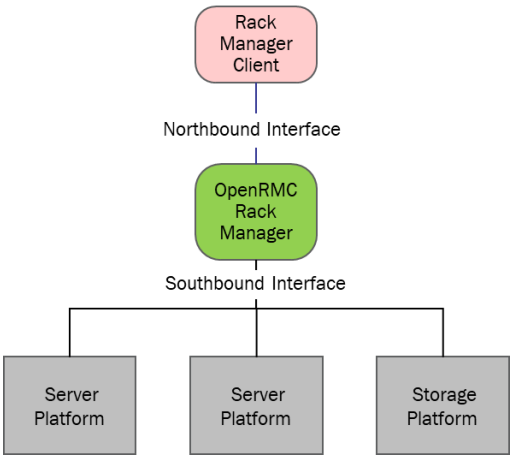

Figure 1 - OpenRMC Rack Manager

The rack manager client communicates with the rack manager service via the northbound interface to manage the platforms within the rack.

The rack manager service manages the platforms within the rack via its southbound interfaces. The platforms could be platforms which supports OCP specified manageability interface or supports other manageability interfaces.

3.1. TERMS

The following are terms used in this document.

The **rack** is an assemblage of multiple trays (or drawers). The rack may contain one or more Power Zones and Thermal Zones.

The **rack management controller** manages the physical aspects of the rack. This includes power, thermal, firmware, etc.

The **node** is a compute, storage or network capability.

The **tray or drawer** is a rack wide sub-chassis which contains one or more nodes.

The **power zone** is a power management domain. The trays/drawers in a power zone share the same power shelf, or the power distribution units (PDU).

The **power shelf** contains power supply units (PSUs)

The **thermal zone** is a thermal management domain. The thermal zone contains one or more trays which zone the same cooling device(s) (aka fan).

4. OpenRMC rack Manager

The OpenRMC firmware shall be referenced by an OCP platform design contribution. Specifically, it shall include OCP OpenRMC repository where the firmware resides. The firmware when built from this repository shall execute on the contributed platform.

The OpenRMC rack manager shall expose a northbound interface which conforms to the northbound interface (section Redfish Specification. The northbound interface shall support the OpenRMC profile. The OpenRMC rack manager may support other northbound interfaces.

The OpenRMC rack manager shall support one or more southbound interfaces which connect the rack management controller to the managed nodes. One of the supported interfaces shall be able to manage the managed node which supports the OCP Baseline Profile and OCP Server Profile.

The OpenRMC rack manager firmware shall be contributed in source form as part of the OCP contribution. The RMC firmware shall be placed in an OCP repository. The OpenRMC firmware source code contribution shall include a README which describes how to build the firmware image.

5. Physical Platform

The OpenRMC Service firmware is hosted on the rack management controller (RMC)

The rack management controller can be contained on various platforms and form-factors. Figure 2 shows the rack manager controller is these various platforms:

- Within the power shelf
- Within the network switch
- On a dedicated sled or tray
- Other form-factors

The design files for the platforms shall be contributed via the appropriate OCP projects.

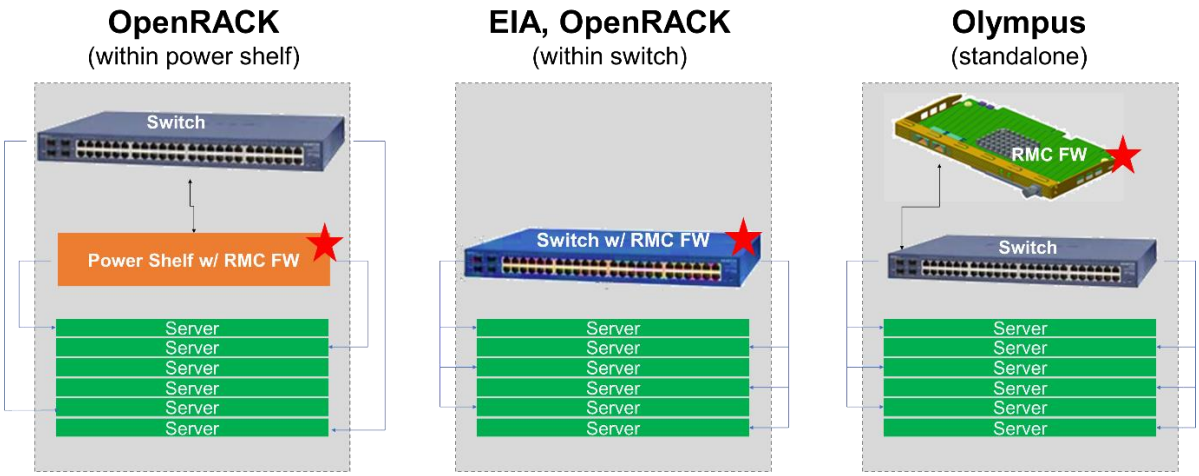

Figure 2 – Rack Management Controller implementations

5.1. CAPABILITIES

The following use cases and associated resources have been identified to allow RMC interface to expose the rack level manageable capabilities.

Use Case	Manageable Capabilities

Hardware inventory	• Get the position and detail of the managed nodes
	  • FRU information of rack manager
	  • FRU information of each node

Rack Power Status	• Obtain the rack power readings
	  • Voltage
	  • Current

Rack Power Control	• Set the rack power usage limit
PSU Status/Health	on/off/capping

Node Power Status	• Determine the power status of the node
	  • On or Off
	• Obtain the node power readings
	  • Voltage
	  • Current

Node Power Control	Power profile
Node Temperature	• Obtain the node temperature • Celsius or Fahrenheit

Node Status/Health	• Obtain the status and health of the node
	  • Status and health of the CPUs
	  • Status and health of the memory
	• Obtain the state of the LED
	• Retrieve the logs

Firmware Versions	• Obtain the firmware revision of
	  • Rack Management firmware
	  • BIOS firmware of each node
	  • BMC firmware of each node
	  • PSU firmware

Firmware Update	Update the firmware on the • Rack Management firmware

Account Management	Admin/user accounts

6. Rack Management Controller Northbound Interface

The OpenRMC interface implementation will support:

- The Redfish interface
- The OpenRMC Profile

The Redfish Interface support is specified in “Redfish API Specification v1.6.0.” The document specifies the behavior of the RESTful interface.

The OpenRMC Profile is a JSON-formatted document which specifies the Redfish resources and resource properties that are required to be supported. The following sections describe the Redfish resources and includes fragments of the OpenRMC Profile.

Conformance to these standards can be determine by running the Redfish test suite with is composed of:

- Redfish Service Validator
- Redfish Service Conformance Check
- Redfish Interop Validator

The Rack Management Controller interface shall support the following resources. The schema for the resources is based on Redfish schema 2018.3.

Table 5. Resource URI (Uniform Resource Identifier)

Resource	URI
Service Root	/redfish/v1
Chassis Collection	/redfish/v1/Chassis
Chassis	/redfish/v1/Chassis/{ID}
Power	/redfish/v1/Chassis/{ID}/Power
Thermal	/redfish/v1/Chassis/{ID}/Thermal
Manager Collection	/redfish/v1/Managers
Managers	/redfish/v1/Managers/{ID}
Network Protocol	/redfish/v1/Managers/{ID}/NetworkProtocol
Ethernet Interfaces Collection	/redfish/v1/Managers/{ID}/EthernetInterfaces
Ethernet Interfaces	/redfish/v1/Managers/{ID}/EthernetInterfaces/{ID}
System Collection	/redfish/v1/Systems
System	/redfish/v1/Systems/{ID}
EventService	/redfish/v1/EventService
Event Subscriptions Collection	/redfish/v1/EventService/Subscriptions
Event Subscription	/redfish/v1/EventService/Subscriptions/{subscriptionID}
TaskService	/redfish/v1/TaskService
Tasks Collection	/redfish/v1/TaskService/Tasks
Tasks	/redfish/v1/TaskService/Tasks/{taskID}
TelemetryService	/redfish/v1/TelemetryService
MetricDefinitions Collection	/redfish/v1/TelemetryService/MetricDefinitions
MetricDefinitions	/redfish/v1/TelemetryService/MetricDefinitions/{metricDefinitionId}
UpdateService	/redfish/v1/UpdateService
SessionService	/redfish/v1/SessionService

LogEntry	/redfish/v1/Managers/{ID}/LogServices/{ID}/Entries/{ID} /redfish/v1/Systems/{ID}LogServices/{ID}/Entries/{ID}
LogService	/redfish/v1/Managers/{ID}/LogServices/{ID} /redfish/v1/Systems/{ID}/LogServices/{ID}

7. Rack Management Controller Resources

This section specifies each of the resources supported by the RMC interfaces and the expected interface behavior.

7.1. Service Root

Service Root resource is the entry point to the Redfish interface. The following is the profile fragment for the resource.

```
"ServiceRoot":{
    "Purpose": "Entry point for the whole API. It contains ",
    "PropertyRequirements": {
        "RedfishVersion": {},
        "UUID": {},
        "AccountService": {},
        "Chassis": {},
        "EventService": {},
        "Managers": {},
        "Registries": {},
        "SessionService": {},
        "Tasks": {},
        "TelemetryService": {},
        "UpdateService": {}
    }
}
```

7.2. Managers Collection Resource

The Managers collection resource contains a list of managers. The following is profile fragment for the ManagerCollection resource.

```
 "ManagerCollection": {
    "PropertyRequirements": {
        "Members": {
            "MinCount": 1
        }
    }
}
```

7.3. Manager Resource

The Manager resource represents the rack manager controller. The following is profile fragment for the Manager resource.

```
"Manager": {
    "Purpose": "Represents the rack management controller.",
    "PropertyRequirements": {
        "ManagerType": {
```

```
                "Values": ["RackManager"]
            },
            "Description": {},
            "UUID": {},
            "ServiceEntryPointUUID": {},
            "Model": {},
            "DateTime": {},
            "DateTimeLocalOffset": {},
            "FirmwareVersion": {},
            "PowerState": {},
            "SerialConsole": {},
            "Status": {
                "PropertyRequirements": {
                    "State": {},
                    "Health": {},
                    "HealthRollup": {}
                }
            },
            "EthernetInterfaces": {
                "Purpose": "Reference to Ethernet interface."
            },
            "ManagerNetworkProtocol": {
                "Purpose": "Reference to supported network protocols."
            },
            "LogServices": {},
            "Links": {
                "PropertyRequirements": {
                    "ManagerForChassis": {
                        "Purpose": "Associates rack manager to managed rack."
                    },
                    "ManagerInChassis": {
                        "Purpose": "Association to the chassis which contains the rack manager."
                    }
                }
            }
        },
        "ActionRequirements": {
            "Reset": {
                "Purpose": "Ability to reset to RMC.",
                "Parameters": {
                    "ResetType": {
                        "MinSupportValues": ["PowerCycle"]
                    }
                }
            }
        }
}
```

7.4. MANAGER NETWORK PROTOCOL RESOURCE

This resource represents the rack management controller's network protocol. The following is profile fragment for the resource.

```
"ManagerNetworkProtocol":{
    "Purpose": "Network protocols supported by the rack manager.",
    "PropertyRequirements": {
        "Status": {},
        "HTTP": {
            "PropertyRequirements": {
                "ProtocolEnable": {},
                "Port": {}
            }
        },
        "HTTPS": {
            "PropertyRequirements": {
                "ProtocolEnable": {},
                "Port": {}
            }
        }
    }
}
```

7.5. MANAGER ETHERNET INTERFACE COLLECTION

This resource represents the collection of rack management controller's Ethernet interfaces. The following is profile fragment for the resource.

```
 "EthernetInterfaceCollection": {
    "PropertyRequirements": {
        "Members": {
            "MinCount": 1
        }
    }
}
```

7.6. ETHERNET INTERFACE

This resource represents the Ethernet interface. The following is profile fragment for the resource.

Note there are conditional requirements that only apply when the Ethernet Interface is subordinate to ./Managers/{id}/EthernetInterfaces.

```
"EthernetInterface": {
    "Purpose": "Manager's Ethernet Interface",
    "MinVersion": "1.1.0",
    "ReadRequirement": "Recommended",
    "ConditionalRequirements": [
        {
            "SubordinateToResource": [
                "Manager",
                "EthernetInterfaceCollection"
            ],
            "ReadRequirement": "Mandatory"
        }
    ],
    "PropertyRequirements": {
        "MACAddress": {},
        "SpeedMbps": {},
```

```
        "InterfaceEnabled": {},
        "LinkStatus": {},
        "Status": {
            "PropertyRequirements": {
                "Health": {},
                "State": {}
            }
        },
        "DHCPv4": {
            "ReadRequirement": "Recommended",
            "WriteRequirement": "Recommended"
        },
        "DHCPv6": {
            "ReadRequirement": "Recommended",
            "WriteRequirement": "Recommended"
        },
        "HostName": {
            "ReadRequirement": "Recommended",
            "ConditionalRequirements": [
                {
                    "SubordinateToResource": [
                        "Manager",
                        "EthernetInterfaceCollection"
                    ],
                    "ReadRequirement": "Mandatory",
                    "WriteRequirement": "Mandatory"
                }
            ]
        },
        "FQDN": {
            "ReadRequirement": "Recommended",
            "ConditionalRequirements": [
                {
                    "SubordinateToResource": [
                        "Manager",
                        "EthernetInterfaceCollection"
                    ],
                    "ReadRequirement": "Mandatory",
                    "WriteRequirement": "Mandatory"
                }
            ]
        },
        "NameServers": {
            "ReadRequirement": "Recommended",
            "ConditionalRequirements": [
                {
                    "SubordinateToResource": [
                        "Manager",
                        "EthernetInterfaceCollection"
                    ],
                    "ReadRequirement": "Mandatory",
                    "WriteRequirement": "Mandatory"
                }
```

```
            ]
        },
        "IPv4Addresses": {
            "ReadRequirement": "Recommended",
            "ConditionalRequirements": [
                {
                    "SubordinateToResource": [
                        "Manager",
                        "EthernetInterfaceCollection"
                    ],
                    "ReadRequirement": "Mandatory",
                    "WriteRequirement": "Mandatory"
                }
            ],
            "PropertyRequirements": {
                "Address": {},
                "SubnetMask": {},
                "AddressOrigin": {},
                "Gateway": {}
            }
        },
        "IPv4StaticAddresses": {
            "ReadRequirement": "Recommended"
        },
        "IPv6AddressPolicyTable": {
            "ReadRequirement": "Recommended"
        },
        "IPv6StaticAddresses": {
            "ReadRequirement": "Recommended"
        },
        "IPv6StaticDefaultGateways": {
            "ReadRequirement": "Recommended"
        },
        "IPv6Addresses": {
            "ReadRequirement": "IfImplemented",
            "PropertyRequirements": {
                "Address": {},
                "PrefixLength": {},
                "AddressOrigin": {},
                "AddressState": {}
            }
        },
        "StaticNameServers": {
            "ReadRequirement": "Recommended"
        }
    }
}
```

7.7. CHASSIS COLLECTION RESOURCE

This resource represents the collection of chassis. The following is profile fragment for the resource.

```
 "ChassisCollection":{
```

```
    "PropertyRequirements": {
        "Members": {
            "MinCount": 1
        }
    }
}
```

7.8. CHASSIS RESOURCE

This resource represents the chassis. The following is profile fragment for the resource

```
"Chassis": {
    "Purpose": "Represents the physical rack.",
    "PropertyRequirements": {
        "ChassisType": {
            "Values": ["Rack"]
        },
        "AssetTag": {},
        "Manufacturer": {},
        "Model": {},
        "PartNumber": {},
        "PowerState": {},
        "SerialNumber": {},
        "Status": {
            "PropertyRequirements": {
                "State": {},
                "Health": {}
            }
        },
        "Thermal": {
            "Purpose": "Reference to subordinate Thermal resource."
        },
        "Power": {
            "Purpose": "Reference to subordinate Power resource."
        },
        "Links": {
            "PropertyRequirements": {
                "ComputerSystems": {},
                "ContainedBy": {},
                "Contains": {},
                "ManagedBy": {},
                "ManagersInChassis": {
                    "Purpose": "Used to create physical topology and relations."
                }
            }
        }
    },
    "ActionRequirements": {
        "Reset": {
            "Purpose": "To control power state of the Rack.",
            "Parameters": {
                "ResetType": {
                    "MinSupportValues": [
```

```
                        "ForceRestart"
                    ]
                }
            }
        }
    }
}
```

7.9. POWER RESOURCE

This resource represents the power domain of a chassis. The following is profile fragment for the resource.

```
"Power":{
    "Purpose": "Represents a power domain.",
    "PropertyRequirements": {
        "Status": {},
        "PowerConsumedWatts": {},
        "PowerControl": {
            "PropertyRequirements": {
                "Status": {},
                "PhysicalContext": {},
                "PowerConsumedWatts": {},
                "PowerMetrics": {
                    "Purpose": "Obtain power statistics.",
                    "PropertyRequirements": {
                        "MinConsumedWatts": {},
                        "MaxConsumedWatts": {},
                        "AverageConsumedWatts": {}
                    }
                },
                "PowerLimit": {
                    "Purpose": "Set power capping limit.",
                    "PropertyRequirements": {
                        "LimitInWatts": {
                            "ReadRequirement": "Mandatory",
                            "WriteRequirement": "Mandatory"
                        }
                    }
                }
            }
        },
        "Voltages": [
            {
                "PropertyRequirements": {
                    "Name": "VRM1 Voltage",
                    "SensorNumber": {},
                    "Status": {},
                    "ReadingVolts": {},
                    "PhysicalContext": "VoltageRegulator"
                }
            }
        ],
```

```
        "PowerSupplies": {
            "Purpose": "Needed for inventory. Min info - N Watt DC power supply",
            "PropertyRequirements": {
                "Status": {},
                "PowerCapacityWatts": {},
                "PowerSupplyType": {},
                "LineInputVoltage": {
                    "ReadRequirement": "Recommended"
                },
                "Model": {
                    "ReadRequirement": "Recommended"
                },
                "Manufacturer": {
                    "ReadRequirement": "Recommended"
                },
                "FirmwareVersion": {
                    "ReadRequirement": "Recommended"
                },
                "SerialNumber": {
                    "ReadRequirement": "Recommended"
                },
                "PartNumber": {
                    "ReadRequirement": "Recommended"
                }
            }
        },
        "Redundancy": {
            "Purpose": "Describe redundant power supply sets"
        }
    }
}
```

7.10. THERMAL RESOURCE

This resource represents the thermal domain of a chassis. The following is profile fragment for the resource.

For temperature readings, the temperatures are required to for each CPU and the inlet airflow. For fans, their health is required.

```
"Thermal":{
    "Purpose": "Represents a cooling domain.",
    "PropertyRequirements": {
        "Status": {},
        "Temperatures": {
            "PropertyRequirements": {
                "Name": {
                    "Values": [
                        "CPU1 Temp",
                        "CPU2 Temp",
                        "Chassis Intake Temp"
                    ]
                },
                "Status": {},
```

```
                "ReadingCelsius": {},
                "PhysicalContext": {}
            }
        },
        "Fans": {
            "PropertyRequirements": {
                "PhysicalContext": {},
                "HotPluggable": {},
                "Status": {
                    "PropertyRequirements": {
                        "Health": {}
                    }
                }
            }
        }
    }
}
```

7.11. SYSTEM COLLECTION RESOURCE

This resource represents the collection of computer systems. The following is the profile fragment for the resource.

```
 "ComputerSystemCollection": {
     "PropertyRequirements": {
          "Members": {
               "MinCount": 1
          }
     }
}
```

7.12. COMPUTER SYSTEM RESOURCE

This resource represents a computer system. The following is the profile fragment for the resource. The UUID is required for each computer system, and the SerialNumber is optional.

```
"ComputerSystem":{
    "Purpose": "Used to manage the computer system (or server).",
    "PropertyRequirements": {
        "Status": {},
          "UUID": {},
        "SerialNumber": {},
        "BIOSVersion": {},
        "ProcessorSummary": {},
        "MemorySummary": {},
        "PowerState": {},
        "Boot": {
            "Purpose": "Used to control computer system's BIOS boot options.",
            "PropertyRequirements": {
                "BootSourceOverrideEnabled": {},
                "BootSourceOverrideTarget": {},
                "BootSourceOverrideMode": {}
            }
        },
```

```
        "EthernetInterfaces": {
            "Purpose": "Reference to Ethernet interfaces"
        },
        "LogEntry": {
            "Purpose": "Reference to LogEntry."
        },
        "LogService": {
            "Purpose": "Reference to LogService."
        },
        "Links": {
            "PropertyRequirements": {
                "Chassis": {
                    "Purpose": "Associates system to rack or tray."
                },
                "ManagerBy": {
                    "Purpose": "Associates system to tray manager (BMC)."
                }
            }
        }
    },
    "ActionRequirements": {
        "Reset": {
            "Purpose": "Used to control power state of the computer system.",
            "Parameters": {
                "ResetType": {
                    "MinSupportValues": [
                        "On",
                        "PowerCycle"
                    ]
                }
            }
        }
    }
}
```

7.13. SESSIONSERVICE

This resource represents the service which shows the sessions supported by the Redfish service. The following is the profile fragment for the resource.

```
"SessionService":{
    "Purpose": "Used to manage sessions service.",
    "PropertyRequirements": {
        "Status": {
            "State": {},
            "Health": {}
        },
        "ServiceEnabled": {},
        "SessionTimeout": {},
        "Sessions": {}
    }
}
```

7.14. TASKSERVICE

This resource represents the service which shows the tasks spawned by the Redfish service. The following is the profile fragment for the resource.

```
"TaskService":{
    "Purpose": "Used to manage Task service.",
    "PropertyRequirements": {
        "DateTime": {},
        "CompletedTaskOverWritePolicy": {},
        "LifeCycleEventOnTaskStateChange": {},
        "Status": {
            "State": {},
            "Health": {}
        },
        "ServiceEnabled": {},
        "Tasks": {}
    }
}
```

7.15. UPDATESERVICE

This resource represents the service which updates the software and firmware images on the system. The following is the profile fragment for the resource.

Note: *In this specification, only the Manager Resources can be updated.*

```
"UpdateService": {
    "Purpose": "Used to manage Update service.",
    "PropertyRequirements": {
        "Status": {
            "State": {},
            "Health": {},
            "HealthRollup": {}
        },
        "ServiceEnabled": {},
        "HttpPushUri": {},
        "FirmwareInventory": {},
        "SoftwareInventory": {}
    },
    "ActionRequirements": {
        "Purpose": "Used to control Update service.",
        "SimpleUpdate": {
            "Parameters": {
                "ImageURI": {
                    "Purpose": "The URI of the software image to be installed."
                },
                "Targets": {
                    "Purpose": "The array of URIs indicating where the update image is to be applied."
                },
                "TransferProtocolType": {
                    "MinSupportValues": [
                        "CIFS",
                        "FTP",
                        "HTTP",
```

```
                        "HTTPS",
                        "SCP",
                        "NFS"
                    ]
                }
            }
        }
    }
}
```

7.16. LogEntry Collection

This resource represents the collection of LogEntries. The following is the profile fragment for the resource.

```
"LogEntryCollection": {
    "PropertyRequirements": {
        "Members": {
            "MinCount": 1
        }
    }
}
```

7.17. LogEntry

This resource defines the record format for a log. It is designed to be used for SEL logs (from IPMI) as well as Event Logs formats. The EntryType field indicates the type of log and the resource includes several additional properties dependent on the EntryType.

```
"LogEntry": {
    "Purpose": "Used to manage Log Entry.",
    "PropertyRequirements": {
        "EntryType": {
            "Values": [
                "Event",
                "SEL"
            ]
        },
        "Severity": {},
        "Created": {},
        "SensorNumber": {},
        "Message": {},
        "MessageId": {},
        "MessageArgs": [],
        "Links": {}
    }
}
```

7.18. LogService Collection

This resource represents the collection of LogServices. The following is the profile fragment for the resource.

```
"LogServiceCollection": {
    "PropertyRequirements": {
```

```
        "Members": {
            "MinCount": 1
        }
    }
}
```

7.19. LOGSERVICE

This resource represents the log service for the resource or service to which it is associated.

```
"LogService":{
    "Purpose": "Used to manage System Log Service.",
    "PropertyRequirements": {
        "MaxNumberOfRecords": {},
        "OverWritePolicy": {},
        "DateTime": {},
        "DateTimeLocalOffset": {},
        "ServiceEnabled": {},
        "LogEntryType": "{}",
        "Status": {
            "State": {},
            "Health": {}
        },
        "Entries": {}
    },
    "ActionRequirements": {
        "Purpose": "Used to clear the log.",
        "ClearLog": {}
    }
}
```

8. REFERENCES

[1] “Redfish API Specification”, 2018
https://www.dmtf.org/sites/default/files/DSP0266_1.6.0.pdf

[2] “Redfish Schema Bundle 2018.1”, 2018,
https://www.dmtf.org/sites/default/files/standards/documents/DSP8010_2018.1.zip

[3] “OCP Baseline Hardware Management Profile v1.0.0”, 2018,
https://github.com/opencomputeproject/OCP-Profiles

[4] “OCP Server Management Profile v0.2.0”, 2018
https://github.com/opencomputeproject/OCP-Profiles

9. REVISION

Revision	Date	Description
1.0.0	July 15, 2020	Final draft