Redfish API for UFM

July 20, 2020

Samsung Semiconductor, Inc.

|  |  |  |
| --- | --- | --- |
| C:\Users\harry.rogers\Pictures\Art\samsung_logo.gif |  |  |

**Table of Contents**

[1 Introduction 5](#_Toc46152979)

[2 List of supported APIs 5](#_Toc46152980)

[3 Diagrams 6](#_Toc46152981)

[4 REST APIs 8](#_Toc46152982)

[5 How to Obtain subsystem information through Redfish API 25](#_Toc46152983)

**Revision History**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rev. | Date | Section | Author | Change |
| 0.1 | 07/20/2020 | All | Lu Fan | Created |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**List of Figures**

No table of figures entries found.

**List of Tables**

**No table of figures entries found.**

**Internal Reference Documents**

|  |  |
| --- | --- |
| Document Info | Title of Document |
|  |  |

# Introduction

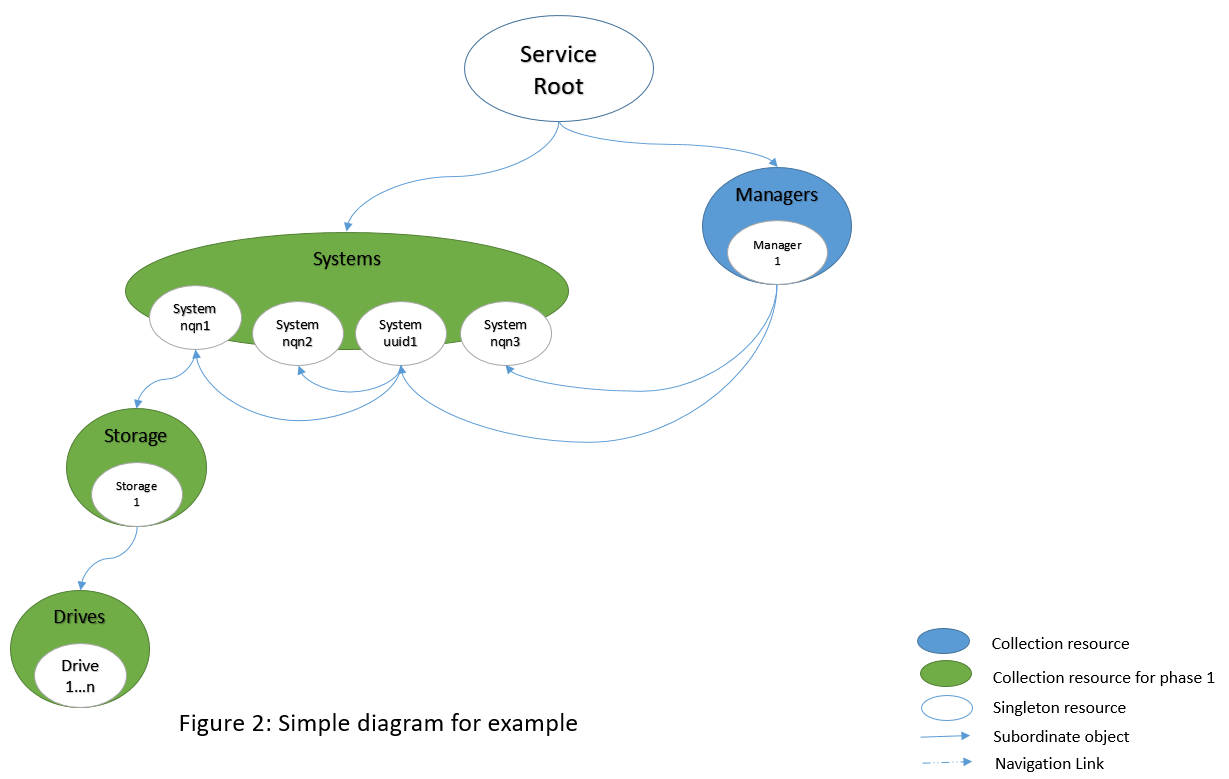
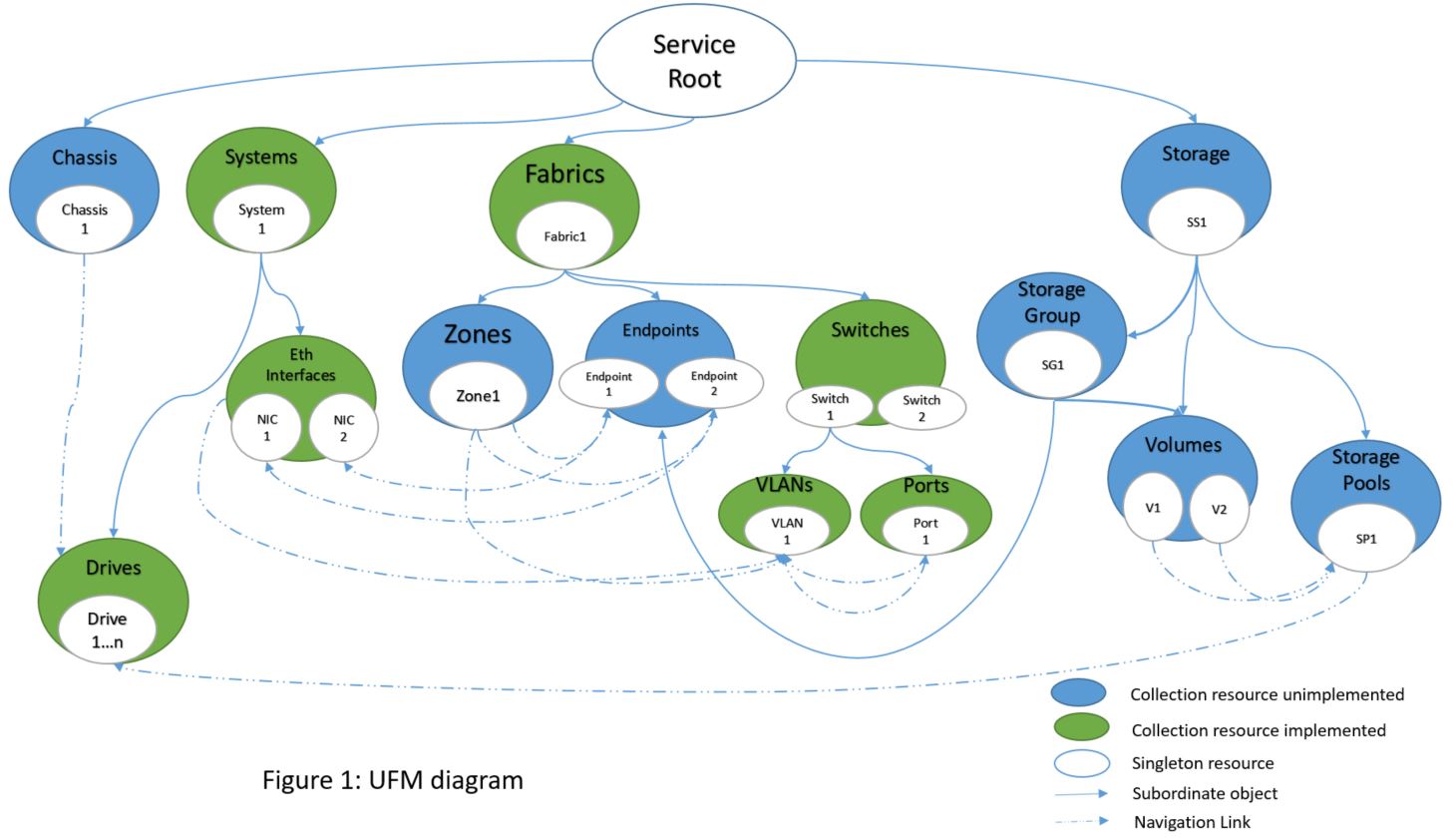
The purpose of this document is to identify all the APIs for management of fabric and storage within UFM.

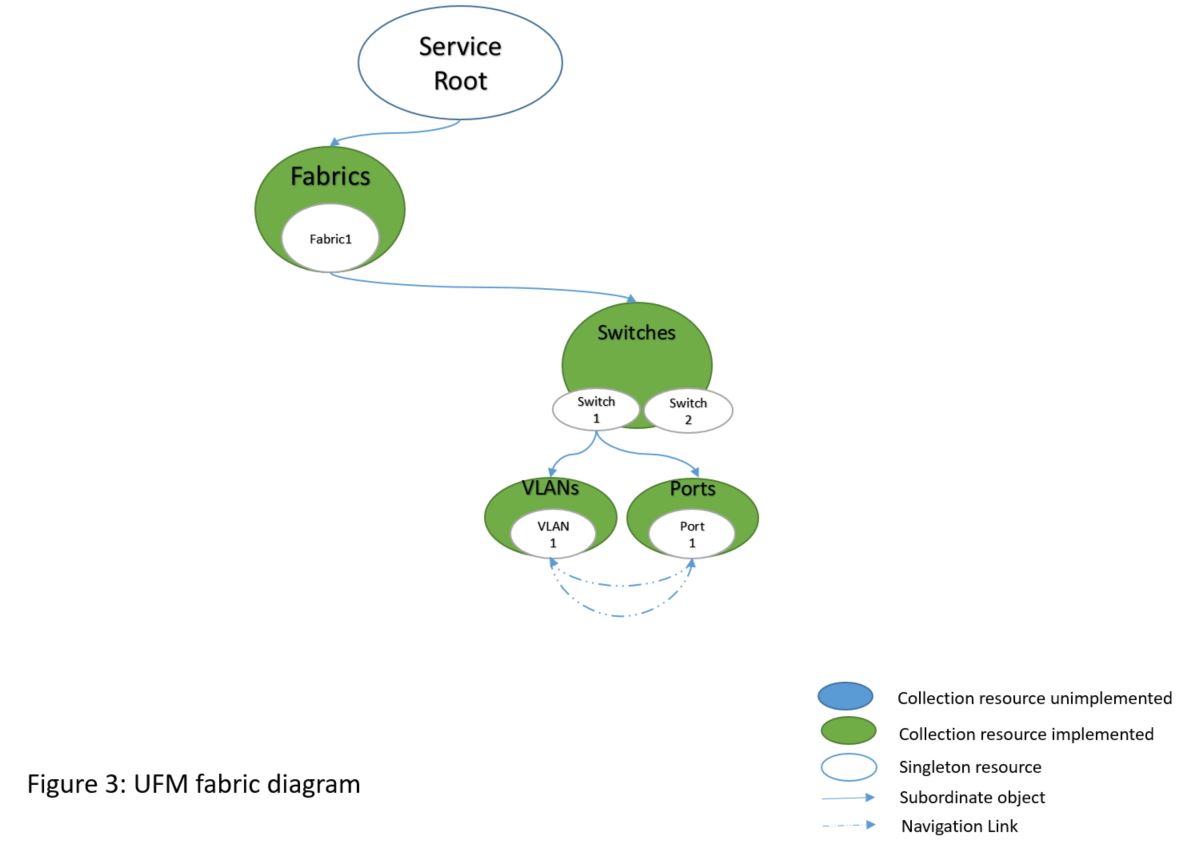
Currently compliant to the Redfish 2020.2. <https://www.dmtf.org/sites/default/files/standards/documents/DSP0266_1.11.0.pdf>

# List of supported APIs

|  |  |
| --- | --- |
| URI | Resource |
| /redfish/v1 | Service Root |
| /redfish/v1/Systems | System Collection |
| /redfish/v1/Systems/{SystemdID} | System Information |
| /redfish/v1/Systems/{SystemdID}/Storage | Storage Collection |
| /redfish/v1/Systems/{SystemdID}/Storage/{StorageID} | Storage Information |
| /redfish/v1/Systems/{SystemdID}/Storage/{StorageID}/Drives | Drive Collection |
| /redfish/v1/Systems/{SystemdID}/Storage/{StorageID}/Drives/{DriveID} | Drive Information |
| /redfish/v1/Systems/{SystemdID}/Storage/{StorageID}/Drives/{DriveID}/Oem/  Samsung/DriveEthernetExt | Samsung DriveEthernet information |
| /redfish/v1/Systems/{SystemdID}/EthernetInterfaces | Network Collection |
| /redfish/v1/Systems/{SystemdID}/EthernetInterfaces/{InterfaceID} | Network Information |
| /redfish/v1/Fabrics | Fabric Collection |
| /redfish/v1/Fabrics/Fabric.1 | Fabric Information |
| /redfish/v1/Fabrics/Fabric.1/Switches | Switch Collection |
| /redfish/v1/Fabrics/Fabric.1/Switches/{SwitchId} | Switch Information |
| /redfish/v1/Fabrics/Fabric.1/Switches/{SwitchId}/Ports | Port Collection |
| /redfish/v1/Fabrics/Fabric.1/Switches/{SwitchId}/Ports/{PortId} | Port Information |
| /redfish/v1/Fabrics/Fabric.1/Switches/{SwitchId}/VLANs | VLAN Collection |
| /redfish/v1/Fabrics/Fabric.1/Switches/{SwitchId}/VLANs/{VLANId} | VLAN Information |

# Diagrams

Figure 1 is the architecture overview. Figure 2 is what the example in section 4 will represent. Figure 3 is the UFM fabric diagram. 



# REST APIs

The service root URL for Redfish version 1.x services will be http://<FabricManager\_IP>/redfish/v1

Returns the following JSON output

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#ServiceRoot.ServiceRoot",    **"@odata.id"**:"/redfish/v1",    **"@odata.type"**:"#ServiceRoot.v1\_0\_0.ServiceRoot",    **"Id"**:"RootService",    **"Name"**:"Root Service",  **"ProtocolFeaturesSupported"**:{  **"ExpandQuery"**:{  **"ExpandAll"**:false  },  **"SelectQuery"**:false  },    **"RedfishVersion"**:"1.8.0",  **"UUID"**:"1f7eb191-e6e3-5ff9-bd55-dc74a62c1b0e",    **"Vendor"**:"Samsung",  **"JsonSchemas"**:{        **"@odata.id"**:"/redfish/v1/JSONSchemas"    },  **"Systems"**:{        **"@odata.id"**:"/redfish/v1/Systems"    },    **"Managers"**:{        **"@odata.id"**:"/redfish/v1/Fabrics"    } } |

http://<FabricManager\_IP>/redfish/v1/Systems

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#ComputerSystemCollection.ComputerSystemCollection",    **"@odata.id"**:"/redfish/v1/Systems",    **"@odata.type"**:"#ComputerSystemCollection.ComputerSystemCollection",    **"Description"**:"Collection of Computer Systems",    **"Members"**:[        {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1"       },       {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid2"       },       {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid2.nqnuuid3"       },       {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1"       },       {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid2"       }    ],    **"Members@odata.count"**:4,    **"Name"**:"System Collection" } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#ComputerSystem.ComputerSystem",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1",    **"@odata.type"**:"#ComputerSystem.v1\_9\_0.ComputerSystem",    **"Description"**:"System containing two subsystems",  **"Id"**:"Targetuuid1",  **"Identifiers":** [  {         **"DurableName"**:" 4d017a48-e3c5-11e9-9efe-44a842481efb",          **"DurableNameFormat"**:"UUID"    }  ],    **"IPv4Addresses"**:[        {           **"Address"**:"10.1.51.54",           **"SubnetMask"**:"",          **"AddressOrigin"**:"",          **"Gateway"**:""       }    ],    **"IPv6DefaultGateway"**:""    **"IPv6Addresses"**:[        {           **"Address"**:"",          **"PrefixLength"**:0,          **"AddressOrigin"**:"",          **"AddressState"**:""       }    ]    **"Links"**:{  “**SupplyingComputerSystems**”: [  {  **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1",  **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid2"  }  ]    }  } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#ComputerSystem.ComputerSystem",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1",    **"@odata.type"**:"#ComputerSystem.v1\_9\_0.ComputerSystem",    **"Description"**:"System representing the drive resources",  **"Id"**:"Targetuuid1.nqnuuid1",  **"Identifiers":** [  {         **"DurableName"**:"nqn.2019-07.samsung:msl-ssg-mp03-data1",           **"DurableNameFormat"**:"NQN"    }  ],    **"Status"**:{        **"State"**:"Enabled",       **"Health"**:"OK"    },  **"Storage"**:{        **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage"    },    **"EthernetInterfaces"**:{        **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces"    },  **"Name"**:"System",    **"Links"**:{  “**ConsumingComputerSystems**”: [  {  **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1"  }  ]    }  **"oem"**:{        **"ServerName"**:"servername1"  **"NSID"**:1  **"NumaAligned"**:true     }  } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#StorageCollection.StorageCollection",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage",    **"@odata.type"**:"#StorageCollection.StorageCollection",    **"Description"**:"Collection of Storage information",    **"Members"**:[        {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1"       }    ],    **"Members@odata.count"**:1,    **"Name"**:"Storage Collection"  **"oem"**:{        **"CapacityBytes"**:8589934592  **"PercentAvailable"**:85     } } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#Storage.Storage",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1",    **"@odata.type"**:"#Storage.v1\_8\_0.Storage",  **"Id"**:"Storage.1",    **"Description"**:"Storage information",    **"Drives"**:[        {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives/ S3VJNA0M835131"       },  {  **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives/ S3VJNA0M835132"  }    ],    **"Name"**:"Storage"  **"oem"**:{        **"CapacityBytes"**:1073741824  **"PercentAvailable"**:80     } } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#DriveCollection.DriveCollection",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives",    **"@odata.type"**:"#DriveCollection.DriveCollection",    **"Description"**:"Collection of drives",    **"Members"**:[        {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives/S3VJNA0M835131"       },        {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives/S3VJNA0M835132"       }     ],    **"Members@odata.count"**:2,    **"Name"**:"Drive Collection"  **"Id"**:"Storage.1" } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives/S3VJNA0M835131

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#Drive.Drive",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/Storage/Storage.1/Drives/S3VJNA0M835131",    **"@odata.type"**:"#Drive.v1\_8\_0.Drive",    **"Description"**:"Drive Information",    **"Name"**:"Storage Drive Information",    **"BlockSizeBytes"**:512,    **"CapacityBytes"**:3840755982336,    **"Id"**:"S3VJNA0M835131",  **"Manufacturer"**:"Samsung",    **"MediaType"**:"SSD",    **"Model"**:"MZQLB3T8HALS-000AZ",  **"Protocol"**:"NVMeOverFabrics",    **"Revision"**:"ETA51KB3",    **"SerialNumber"**:"S3VJNA0M835131",  **"oem"**:{   **"PercentAvailable"**:80     } } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#EthernetInterfaceCollection.EthernetInterfaceCollection",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces",    **"@odata.type"**:"#EthernetInterfaceCollection.EthernetInterfaceCollection",    **"Description"**:"Collection of Ethernet Interfaces",    **"Members"**:[        {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces/NIC.1"       },       {           **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces/NIC.2"       }    ],    **"Members@odata.count"**:2,    **"Name"**:"Ethernet interfaces Collection" } |

http://<FabricManager\_IP>/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces/NIC.1

|  |
| --- |
| {     **"@odata.context"**:"/redfish/v1/$metadata#EthernetInterface.EthernetInterface",    **"@odata.id"**:"/redfish/v1/Systems/Targetuuid1.nqnuuid1/EthernetInterfaces/NIC.1",    **"@odata.type"**:"#EthernetInterface.v1\_5\_1.EthernetInterface",    **"Name"**:"NIC 1",  **"Description"**:"Ethernet Interface information",    **"Id"**:"NIC.1",    **"LinkStatus"**:"LinkUp",  **"MACAddress"**:"aa:bb:cc:dd:ee:ff",  **"SpeedMbps"**:"100000",  **"IPv4Addresses"**:[        {           **"Address"**:"102.100.20.3",          **"SubnetMask"**:"",          **"AddressOrigin"**:"",          **"Gateway"**:""  **"oem"**:{   **"Port"**:1023,  **"SupportedProtocol"**:”TCP”      }       }    ],    **"IPv6DefaultGateway"**:""    **"IPv6Addresses"**:[        {           **"Address"**:"",          **"PrefixLength"**:0,          **"AddressOrigin"**:"",          **"AddressState"**:""  **"oem"**:{   **"Port"**:1023,  **"SupportedProtocol"**:”TCP”    }       }    ] } |

http://<FabricManager\_IP>/redfish/v1/Fabrics

|  |
| --- |
| {   **"@odata.id"**: "/redfish/v1/Fabrics",  **"@odata.type"**: "#FabricCollection.FabricCollection",  **"Description"**: "Collection of Fabrics",  **"Name"**: "Fabric Collection",  **"Members"**: [  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1"  }  ],  **"Members@odata.count"**:1  } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1",  **"@odata.type"**: "#Fabric.v1\_1\_1.Fabric",  **"Id"**: "Fabric.1",  **"Description"**: "Fabric information",  **"Name"**: "Fabric",  **"FabricType"**: "NVME",  **"Switches"**: {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches"  } } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1/Switches

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1/Switches",  **"@odata.type"**: "#SwitchCollection.SwitchCollection",  **"Description"**: "Collection of Switches",  **"Name"**: "Ethernet Switches Collection",  **"Members"**: [  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/cd440a20-828c-11ea-8000-1c34da948340"  }  ],  **"Members@odata.count"**: 2  } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0",  **"@odata.type"**: "#Switch.v1\_3\_1.Switch",  **"Id"**: "ab35df6c-1108-11ea-8000-1c34da32f0c0",  **"Description"**: "Ethernet Switch information",  **"Name"**: "Switch",  **"SerialNumber"**: "MT1947X24003",  **"UUID"**: "ab35df6c-1108-11ea-8000-1c34da32f0c0",  **"Oem"**: {  "PFC": "enabled",  "PriorityEnabledList": [  "3"  ],  "PriorityDisabledList": [  "0",  "1",  "2",  "4",  "5",  "6",  "7"  ]  },  **"Actions"**: {  "#EnablePfcGlobally": {  "description": "Enables priority flow control globally on the switch.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Actions/EnablePfcGlobally"  },  "#DisablePfcGlobally": {  "description": "Disables priority flow control globally on the switch.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Actions/DisablePfcGlobally"  },  "#EnablePfcPerPriority": {  "description": "Enables PFC per priority on the switch.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Actions/EnablePfcPerPriority",  "Parameters": [  {  "Name": "Priority",  "Required": true,  "DataType": "Number",  "MinimumValue": "0",  "MaximumValue": "7"  }  ]  },  "#DisablePfcPerPriority": {  "description": "Disables PFC per priority on the switch.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Actions/DisablePfcPerPriority",  "Parameters": [  {  "Name": "Priority",  "Required": true,  "DataType": "Number",  "MinimumValue": "0",  "MaximumValue": "7"  }  ]  },  "#AnyCmd": {  "description": "Send any cmd directly to the switch.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Actions/AnyCmd",  "Parameters": [  {  "Name": "AnyCmdStr",  "Required": true  }  ]  }  },  **"Ports"**: {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports"  },  **"VLANs"**: {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs"  }  } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1/Switch/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports",  **"@odata.type"**: "#PortCollection.PortCollection",  **"Description"**: "Collection of Ports",  **"Name"**: "Port Collection",  **"Members"**: [  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/10"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/11"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/12"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/13"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/14"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/15"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/16"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/2"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/3"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/4"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/5"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/6"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/7"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/8"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/9"  }  ],  **"Members@odata.count"**: 16  } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1",  **"@odata.type"**: "#Port.v1\_2\_1.Port",  **"Id"**: "1",  **"Description"**: "Port Interface",  **"PortId"**: "1",  **"Name"**: "Eth1/1",  **"Mode"**: "access",  **"Oem"**: {  "PFC": "Disabled",  "TrustMode": "L2"  },  **"Links"**: {  "AccessVLAN": {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/1"  },  "AllowedVLANs": []  },  **"Actions"**: {  "#SetAccessPortVLAN": {  "description": "Set this port to access mode that connects to a host. Must specify a default configured VLAN.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/SetAccessPortVLAN",  "Parameters": [  {  "Name": "VLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  }  ]  },  "#UnassignAccessPortVLAN": {  "description": "Set this port access VLAN to default 1.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/UnassignAccessPortVLAN"  },  "#SetTrunkPortVLANsAll": {  "description": "Set this port to trunk mode connecting 2 switches. By default, a trunk port is automatically a member on all current and future VLANs.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/SetTrunkPortVLANsAll"  },  "#SetTrunkPortVLANsRange": {  "description": "Set this port to trunk mode connecting 2 switches. This trunk port is a member on the range of VLANs specified by RangeFromVLANId and RangeToVLANId.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/SetTrunkPortVLANsRange",  "Parameters": [  {  "Name": "RangeFromVLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  },  {  "Name": "RangeToVLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  }  ]  },  "#SetHybridPortAccessVLAN": {  "description": "Set this port to hybrid mode and specify the access VLAN. ",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/SetHybridPortAccessVLAN",  "Parameters": [  {  "Name": "VLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  }  ]  },  "#SetHybridPortAllowedVLAN": {  "description": "Set this port to hybrid mode and add the allowed VLAN. ",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/SetHybridPortAllowedVLAN",  "Parameters": [  {  "Name": "VLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  }  ]  },  "#RemoveHybridPortAllowedVLAN": {  "description": "Remove the specified VLAN from the hybrid port. ",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/RemoveHybridPortAllowedVLAN",  "Parameters": [  {  "Name": "VLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  }  ]  },  "#EnablePortPfc": {  "description": "Enables PFC mode on port.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/EnablePortPfc"  },  "#DisablePortPfc": {  "description": "Disables PFC mode on port.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/DisablePortPfc"  },  "#EnableEcnMarking": {  "description": "Enables explicit congestion notification for traffic class queue.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/EnableEcnMarking",  "Parameters": [  {  "Name": "TrafficClass",  "Required": true,  "DataType": "Number",  "MinimumValue": "0",  "MaximumValue": "7"  },  {  "Name": "MinAbsoluteInKBs",  "Required": true,  "DataType": "Number"  },  {  "Name": "MaxAbsoluteInKBs",  "Required": true,  "DataType": "Number"  }  ]  },  "#DisableEcnMarking": {  "description": "Disables ECN marking for traffic class queue.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/DisableEcnMarking"  },  "#ShowPfcCounters": {  "description": "Display priority flow control counters for the specified interface and priority.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/ShowPfcCounters",  "Parameters": [  {  "Priority": "VLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "0",  "MaximumValue": "7"  }  ]  },  "#ShowCongestionControl": {  "description": "Displays specific interface congestion control information.",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1/Actions/ShowCongestionControl"  }  }  } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1/Switch/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs",  **"@odata.type"**: "#VLanCollection.VLanCollection",  **"Description"**: "Collection of VLANs",  **"Name"**: "VLANs Collection",  **"Members"**: [  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/1"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/10"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/2"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/3"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/4"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/5"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/6"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/7"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/8"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/9"  }  ],  **"Members@odata.count"**: 10,  **"Actions"**: {  "#CreateVLAN": {  "description": "Create a VLAN with Id",  "target": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/Actions/CreateVLAN",  "Parameters": [  {  "Name": "VLANId",  "Required": true,  "DataType": "Number",  "MinimumValue": "1",  "MaximumValue": "4094"  }  ]  }  }  } |

http://<FabricManager\_IP>/redfish/v1/Fabrics/Fabric.1/Switch/ab35df6c-1108-11ea-8000-1c34da32f0c0/VLANs/1

|  |
| --- |
| {  **"@odata.id"**: "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports",  **"@odata.type"**: "#PortCollection.PortCollection",  **"Description"**: "Collection of Ports",  **"Name"**: "Port Collection",  **"Members"**: [  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/1"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/10"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/11"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/12"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/13"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/14"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/15"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/16"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/2"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/3"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/4"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/5"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/6"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/7"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/8"  },  {  "@odata.id": "/redfish/v1/Fabrics/Fabric.1/Switches/ab35df6c-1108-11ea-8000-1c34da32f0c0/Ports/9"  }  ],  **"Members@odata.count"**: 16  } |

# 

# How to Obtain Subsystem Information through Redfish API

For initial understanding of Redfish, please explore this mockup for Simple Rack-mounted Server with Local Storage: <https://redfish.dmtf.org/redfish/mockups/v1/897>. Below is the Python code to get subsystem information from the Redfish API.

|  |
| --- |
| import redfish\_client  import argparse  import json  from rest\_api.test.clustermap\_utils import get\_transport\_type, get\_percent\_available  '''  Create a connection to the UFM redfish API service using the redfish-client  package.  redfish\_client abstracts the connection and operations to a redfish compliant  service.  The connect call returns the redfish service root.  Client code can then iterate through redfish endpoints hierarchy without  making any explicit rest calls.  '''  def getSystemInfo(service\_addr):  # Will throw exception if connection fails. No need to catch it as nothing more to do  # in that case  root = redfish\_client.connect(service\_addr, '', '')  # If the Systems collection is empty return an empty dictionary  if not root.Systems.Members:  return {}  # Add system info  system\_info\_dict = {}  tgt\_cnt = 0  for system in root.Systems.Members:  # Add nqn for any subsystems  if 'Identifiers' in system:  for identifier in system.Identifiers:  if identifier.DurableNameFormat == 'NQN':  tgt\_cnt += 1  tgt\_str = 'System ' + str(tgt\_cnt)  system\_info\_dict[tgt\_str] = {}  system\_info\_dict[tgt\_str]['NQN'] = identifier.DurableName  system\_info\_dict[tgt\_str]['Target\_Server\_Name'] = system.oem['ServerName']  # Add Network Information  if 'EthernetInterfaces' in system:  # No transports to add since no NICs available  if not system.EthernetInterfaces.Members:  continue  # Add the transport fields for each NIC  ethernet\_list = []  ethernet\_str = ','.join(tuple(('{:^20}'.format("Mac Addr"),  '{:^40}'.format("IP Addr"),  '{:^6}'.format("Port"),  '{:^12}'.format("Trans type"),  '{:^10}'.format("Status"))))  ethernet\_list.append(ethernet\_str)  ethernet\_str = ','.join(tuple(('{:^20}'.format(''.rjust(20,'-')),  '{:^40}'.format(''.rjust(14,'-')),  '{:^6}'.format(''.rjust(6,'-')),  '{:^12}'.format(''.rjust(12,'-')),  '{:^10}'.format(''.rjust(10,'-')))))  ethernet\_list.append(ethernet\_str)  for interface in system.EthernetInterfaces.Members:  if 'IPv4Addresses' in interface:  for ipv4addr in interface.IPv4Addresses:  if ipv4addr.Address:  transport\_type, port = get\_transport\_type(ipv4addr)  ethernet\_str = ','.join(tuple((  '{:^20}'.format(str(interface.MACAddress)),  '{:^40}'.format(str(ipv4addr.Address)),  '{:^6}'.format(str(port)),  '{:^12}'.format(str(transport\_type)),  '{:^10}'.format(str(interface.LinkStatus)))))  ethernet\_list.append(ethernet\_str)  if 'IPv6Addresses' in interface:  for ipv6addr in interface.IPv6Addresses:  if ipv6addr.Address:  transport\_type, port = get\_transport\_type(ipv4addr)  ethernet\_str = ','.join(tuple((  '{:^20}'.format(str(interface.MACAddress)),  '{:^40}'.format(str(ipv6addr.Address)),  '{:^6}'.format(str(port)),  '{:^12}'.format(str(transport\_type)),  '{:^10}'.format(str(interface.LinkStatus)))))  ethernet\_list.append(ethernet\_str)  system\_info\_dict[tgt\_str]['EthernetInterfaces'] = ethernet\_list  # Add storage Information  if 'Storage' in system:  # No storage drives to add  if not system.Storage.Members:  continue  # Add the storage fields for each drive  storage\_list = []  storage\_str = ','.join(tuple(('{:^18}'.format("Serial Number"),  '{:^18}'.format("MediaType"),  '{:^18}'.format("Manufacturer"),  '{:^24}'.format("Model"),  '{:^18}'.format("FW Revision"))))  storage\_list.append(storage\_str)  storage\_str = ','.join(tuple(('{:^18}'.format(''.rjust(18,'-')),  '{:^18}'.format(''.rjust(18,'-')),  '{:^18}'.format(''.rjust(18,'-')),  '{:^24}'.format(''.rjust(24,'-')),  '{:^18}'.format(''.rjust(18,'-')))))  storage\_list.append(storage\_str)  for storage in system.Storage.Members:  if 'Drives' in storage:  for drive in storage.Drives:  storage\_str = ','.join(tuple(('{:^18}'.format(str(drive.SerialNumber)),  '{:^18}'.format(str(drive.MediaType)),  '{:^18}'.format(str(drive.Manufacturer)),  '{:^24}'.format(str(drive.Model)),  '{:^18}'.format(str(drive.Revision)))))  storage\_list.append(storage\_str)  system\_info\_dict[tgt\_str]['Storage'] = storage\_list  return system\_info\_dict  if \_\_name\_\_ == '\_\_main\_\_':  parser = argparse.ArgumentParser(  description='Process Server\'s Configuration Settings.')  parser.add\_argument("--host", help="IP Address or FQDN of Server",  dest="host", required=True)  parser.add\_argument("--port", help="Port of Server",  dest="port", default=5000)  args = parser.parse\_args()  service\_addr = 'http://' + args.host + ':' + str(args.port)  systemInfo = {}  systemInfo = getSystemInfo(service\_addr)  print('System Information : \n', json.dumps(systemInfo, indent=2))} |