Enhanced Out Of Band Performance Monitoring

This Dell Technical white paper provides detailed information about OOB performance monitoring for CPU, Memory and I/O utilization using RACADM, WS-Man and WebGUI Interfaces.

Dell Engineering
January 2015

Sudarshan S
Raveendra Reddy P
Smiti Gupta
Rajesh Patel

THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.

PRODUCT WARRANTIES APPLICABLE TO THE DELL PRODUCTS DESCRIBED IN THIS DOCUMENT MAY BE FOUND AT: http://www.dell.com/learn/us/en/19/terms-of-sale-commercial-and-public-sector Performance of network reference architectures discussed in this document may vary with differing deployment conditions, network loads, and the like. Third party products may be included in reference architectures for the convenience of the reader. Inclusion of such third party products does not necessarily constitute Dell's recommendation of those products. Please consult your Dell representative for additional information.

Trademarks used in this text:

Dell™, the Dell logo, Dell Boomi™, Dell Precision™ ,OptiPlex™, Latitude™, PowerEdge™, PowerVault™, PowerConnect™, OpenManage™, EqualLogic™, Compellent™, KACE™, FlexAddress™, Force10™ and Vostro™ are trademarks of Dell Inc. Other Dell trademarks may be used in this document. Cisco Nexus®, Cisco MDS®, Cisco NX-0S®, and other Cisco Catalyst® are registered trademarks of Cisco System Inc. EMC VNX®, and EMC Unisphere® are registered trademarks of EMC Corporation. Intel®, Pentium®, Xeon®, Core® and Celeron® are registered trademarks of Intel Corporation in the U.S. and other countries. AMD® is a registered trademark and AMD Opteron™, AMD Phenom™ and AMD Sempron™ are trademarks of Advanced Micro Devices, Inc. Microsoft®, Windows®, Windows Server®, Internet Explorer®, MS-DOS®, Windows Vista® and Active Directory® are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Red Hat® and Red Hat® Enterprise Linux® are registered trademarks of Red Hat, Inc. in the United States and/or other countries. Novell® and SUSE® are registered trademarks of Novell Inc. in the United States and other countries. Oracle® is a registered trademark of Oracle Corporation and/or its affiliates. Citrix®, Xen®, XenServer® and XenMotion® are either registered trademarks or trademarks of Citrix Systems, Inc. in the United States and/or other countries. VMware®, Virtual SMP®, vMotion®, vCenter® and vSphere® are registered trademarks or trademarks of VMware, Inc. in the United States or other countries. IBM® is a registered trademark of International Business Machines Corporation. Broadcom® and NetXtreme® are registered trademarks of Broadcom Corporation. Qlogic is a registered trademark of QLogic Corporation. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and/or names or their products and are the property of their respective owners. Dell disclaims proprietary interest in the marks and names of others.

Table of contents

Executive summary

Intel ME(Management Engine) provides support for Compute Usage Per Second (CUPS) functionality for monitoring real-time performance index of CPU, Memory and I/O. It obtains utilization information of these platform resources by querying the data from a set of dedicated counters provided by the CPU and chipset.These operations are independent of OS and do not consume CPU resources. Using this functionality of Intel ME, a user can monitor real-time and historical utilization information through iDRAC interfaces.

This whitepaper aims at providing detailed information about OOB performance monitoring for CPU, Memory, I/O and System utilization using RACADM, WS-Man and Web GUI Interfaces.

1 OOB Performance Monitoring Overview

Dell 13th Generation Server Platforms use Intel ME processors that provide support for CUPS functionality for monitoring performance index of CPU, Memory and I/O. It obtains utilization information of these platform resources by querying the data from a set of dedicated counters provided by the CPU and chipset. These counters are called Resource Monitoring Counters or RMCs. These counters are aggregated by iDRAC to measure cumulative utilization of each of these platform resources.

CPU Utilization: There are individual counters (RMCs) for each CPU core which are aggregated to provide cumulative utilization of all the cores in the platform.

Memory Utilization: There are individual counters (RMCs) to measure memory traffic occurring at each memory channel or memory controller instance. These counters are aggregated to measure the cumulative memory traffic across all the memory channels on the platform. This does not indicate the amount of memory utilization, it is indication of memory bandwidth utilization.

IO Utilization: There are individual counters (RMCs), one per root port in the PCI Express Root Complex to measure PCI Express traffic to/from that root port. These counters are aggregated to measure IO traffic. This index is measure of PCIe bandwidth utilization.

System Utilization: System utilization index is calculated by aggregating CPU, Memory and I/O index. The System Utilization Index is a measure of the compute headroom available on the server.

Hence, if the system has a large System Utilization Index, then there is limited headroom to place additional workload on that system. As the resource consumption decreases, the system Utilization Index decreases. A low Index would indicate that there is a large amount of compute headroom and the server is a prime target for receiving new workloads or having the workload migrated off and the server being put into a lower power state in order to reduce power consumption.

Such workload monitoring can then be applied throughout the data center to provide a high-level and holistic view of the datacenter's workload, providing a truly Dynamic Datacenter solution.

1.1 Detailed Description

This feature involves monitoring of the utilization index for CPU, Memory, IO and overall System Level utilization index. It consists of the below characteristics -

- The utilization indexes are populated as IPMI sensors to iDRAC.
- These sensors are Threshold based sensor, it provides configurable upper warning threshold.
- These sensors are sampled at every 1 minute and aggregated for statistical information.
- The current sensor readings of CPU, Memory and I/O Utilization Index, System Utilization Index can be obtained from from RACADM, WS-Man and Web GUI.
- The upper warning threshold can be configured using RACADM, WS-MAN and Web-GUI.
- The SEL event for upper warning threshold is disabled by default. This events can be enabled using standard IPMI commands.

- Sensor readings are collected and historical data is maintained by iDRAC on a last hourly, daily and weekly basis for CPU, Memory, IO and System Utilization sensors.
  - For hourly historical data, each sample is collected every minute.
  - For daily historical data, each sample is collected every hour.
  - For weekly historical data, each sample is collected every 12 Hours (twice a day).
- The historical data is represented in terms of Average, Minimum and Maximum for last hour, day and week.
- The historical peak value is maintained for CPU, Memory, IO and System Utilization.
  - The peak value is the maximum sensor reading.
  - User can reset Historical Peak values.
- The historical statistics data will be available through interfaces like RACADM, WS-Man and Web GUI.

1.2 Prerequisites

There are following prerequisites must be met for this feature to work.

- 13G platforms which support Intel ME will have this feature supported. Since 12G Node Manager doesn’t support CUPS functionality, this feature will not be available on the 12G platforms.
- An Enterprise software license for 13th generation Dell PowerEdge servers. For more information about managing licenses using iDRAC Web interface, click **Overview -> Server -> Licenses**, and then click Help in the upper-right corner.
  - Statistical Information and Peak Reset require Enterprise Level License.
  - Sensor reading and Threshold Configuration does not require any License.

2 Statistical Information

The statistical information of Utilization Monitoring sensors can be viewed from RACADM and WSMAN using the below sensor FQDD's. The FQDD's for statistics objects are defined as :

Table 1 FQDD of Utilization Statistics

FQDD	Description
iDRAC.Embedded.1#SystemBoardCPUUsageStat	CPU Utilization Statistics
iDRAC.Embedded.1#SystemBoardIOUsageStat	IO Utilization Statistics
iDRAC.Embedded.1#SystemBoardMEMUsageStat	Memory Utilization Statistics
iDRAC.Embedded.1#SystemBoardSYSUsageStat	System Utilization Statistics

2.1 Using RACADM

The statistical information of the Utilization Monitoring sensors can be obtained using the sensor FQDD's defined for viewing the usage statistics.

To view the sensor FQDD's defined for reading the utilization statistics

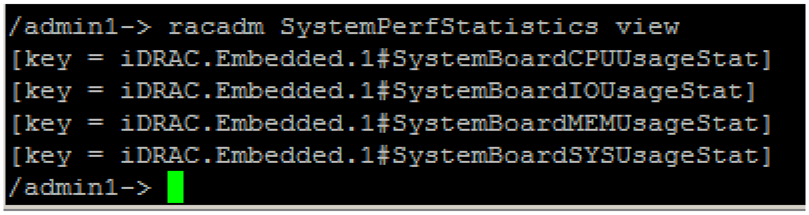
```
racadm SystemPerfStatistics view
```

To view the statistics of the Utilization monitoring sensors:

```
racadm SystemPerfStatistics <Statistics_FQDD>
```

2.1.1 Usage Examples

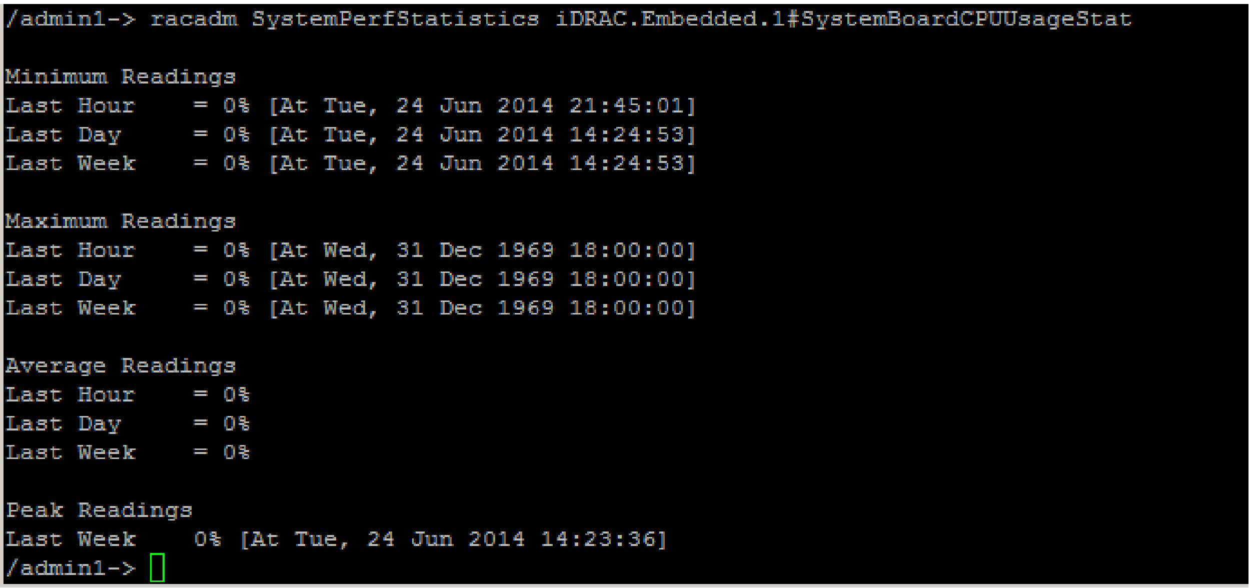
To view the sensor FQDD's defined for reading the utilization statistics



```
/admin1-> racadm SystemPerfStatistics view
[key = iDRAC.Embedded.1#SystemBoardCPUUsageStat]
[key = iDRAC.Embedded.1#SystemBoardIOUsageStat]
[key = iDRAC.Embedded.1#SystemBoardMEMUsageStat]
[key = iDRAC.Embedded.1#SystemBoardSYSUsageStat]
/admin1->
```

To view the Utilization Statistics for the CPU sensor

```
racadm SystemPerfStatistics iDRAC.Embedded.1#SystemBoardCPUUsageStat
```



```
/admin1-> racadm SystemPerfStatistics iDRAC.Embedded.1#SystemBoardCPUUsageStat

Minimum Readings
Last Hour     = 0% [At Tue, 24 Jun 2014 21:45:01]
Last Day      = 0% [At Tue, 24 Jun 2014 14:24:53]
Last Week     = 0% [At Tue, 24 Jun 2014 14:24:53]

Maximum Readings
Last Hour     = 0% [At Wed, 31 Dec 1969 18:00:00]
Last Day      = 0% [At Wed, 31 Dec 1969 18:00:00]
Last Week     = 0% [At Wed, 31 Dec 1969 18:00:00]

Average Readings
Last Hour     = 0%
Last Day      = 0%
Last Week     = 0%

Peak Readings
Last Week     0% [At Tue, 24 Jun 2014 14:23:36]
/admin1->
```

2.2 Using WS-Man

All the statistical data associated to Utilization Sensors are modelled under BaseMetrics profile.

The various categories of statistical data is presented as instances of DCIM_AggregationMetricDefinition class. The instances of this class are: i.e., CPUPerfStatistics, MemoryPerfStatistics, IOPerfStatistics, CUPSIndexStatistics.

The usage statistics calculated over stipulated periods of time like hour, day and week form the values of the DCIM_AggregationMetricValues class. The instances of this class are MinLastHour, MinLastday, MinLastWeek, MaxLastHour, MaxLastday, MaxLastWeek, PeakUtilization, AverageLastHour, AverageLastDay, AverageLastWeek.

The detailed explanation of properties of DCIM_AggregationMetricDefinition and DCIM_AggregationMetricValues class is presented in BaseMetrics Profile Document.

2.2.1 Usage Examples

This section explains WS-Man commands with examples for enumeration of DCIM_AggregationMetricDefinition & DCIM_AggregationMetricValues.

2.2.1.1 Enumerate DCIM_AggregationMetricDefinition

DCIM_AggregationMetricDefinition is a derivation of CIM_AggregationMetricDefinition class.

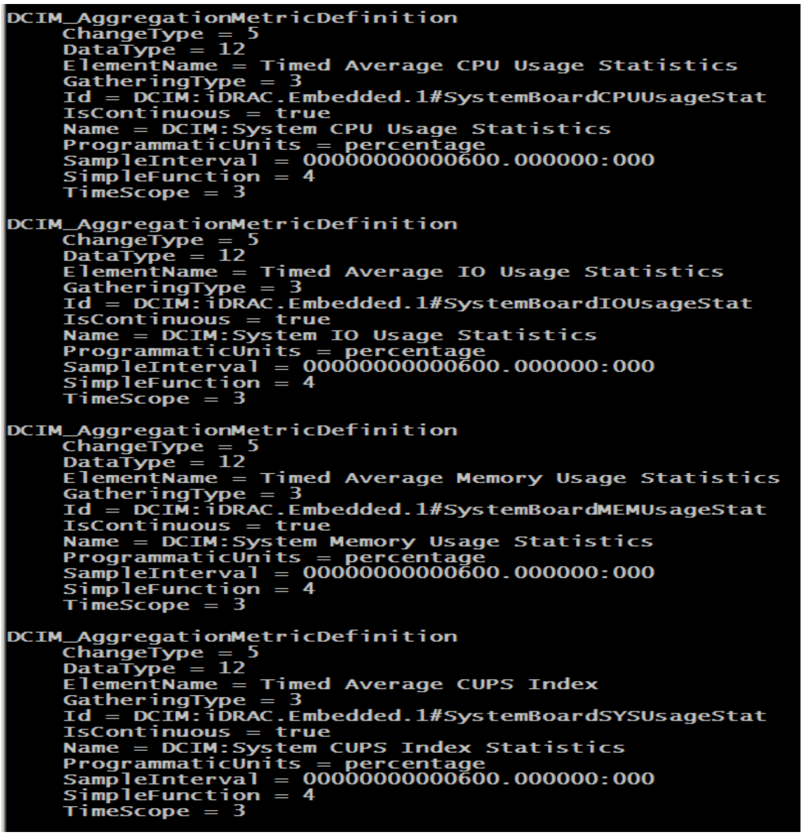
Winrm Command:

```
winrm enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_AggregationMetricDefinition -u:<username> -p:<password> -r:https://<Ip-Address>/wsman -SkipCNcheck -SkipCAcheck -SkipRevocationCheck -encoding:utf-8 -a:basic
```

Note:

- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
DCIM_AggregationMetricDefinition
    ChangeType = 5
    DataType = 12
    ElementName = Timed Average CPU Usage Statistics
    GatheringType = 3
    Id = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    IsContinuous = true
    Name = DCIM:System CPU Usage Statistics
    ProgrammaticUnits = percentage
    SampleInterval = 00000000000600.000000:000
    SimpleFunction = 4
    TimeScope = 3

DCIM_AggregationMetricDefinition
    ChangeType = 5
    DataType = 12
    ElementName = Timed Average IO Usage Statistics
    GatheringType = 3
    Id = DCIM:iDRAC.Embedded.1#SystemBoardIOUsageStat
    IsContinuous = true
    Name = DCIM:System IO Usage Statistics
    ProgrammaticUnits = percentage
    SampleInterval = 00000000000600.000000:000
    SimpleFunction = 4
    TimeScope = 3

DCIM_AggregationMetricDefinition
    ChangeType = 5
    DataType = 12
    ElementName = Timed Average Memory Usage Statistics
    GatheringType = 3
    Id = DCIM:iDRAC.Embedded.1#SystemBoardMEMUsageStat
    IsContinuous = true
    Name = DCIM:System Memory Usage Statistics
    ProgrammaticUnits = percentage
    SampleInterval = 00000000000600.000000:000
    SimpleFunction = 4
    TimeScope = 3

DCIM_AggregationMetricDefinition
    ChangeType = 5
    DataType = 12
    ElementName = Timed Average CUPS Index
    GatheringType = 3
    Id = DCIM:iDRAC.Embedded.1#SystemBoardSYSUsageStat
    IsContinuous = true
    Name = DCIM:System CUPS Index Statistics
    ProgrammaticUnits = percentage
    SampleInterval = 00000000000600.000000:000
    SimpleFunction = 4
    TimeScope = 3
```

Customers can use the programmatic interface of WS-Man to create their own management applications. So SOAP request packet for this class enumeration is given below:

SOAP XML:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/enumeration/Enumerate</wsa:Actio
n>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/DCIM_AggregationMetricDefinition</wsman:ResourceURI>
...
</wsman:SelectorSet>
</s:Header>
<s:Body>
<wsen:Enumerate>
<wsman:OptimizeEnumeration/>
<wsman:MaxElements>32000</wsman:MaxElements>
</wsen:Enumerate>
</s:Body>
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443

2.2.1.2 Get Instance of DCIM_AggregationMetricDefinition

Using get operation only desired instance can be viewed.The command is shown below:

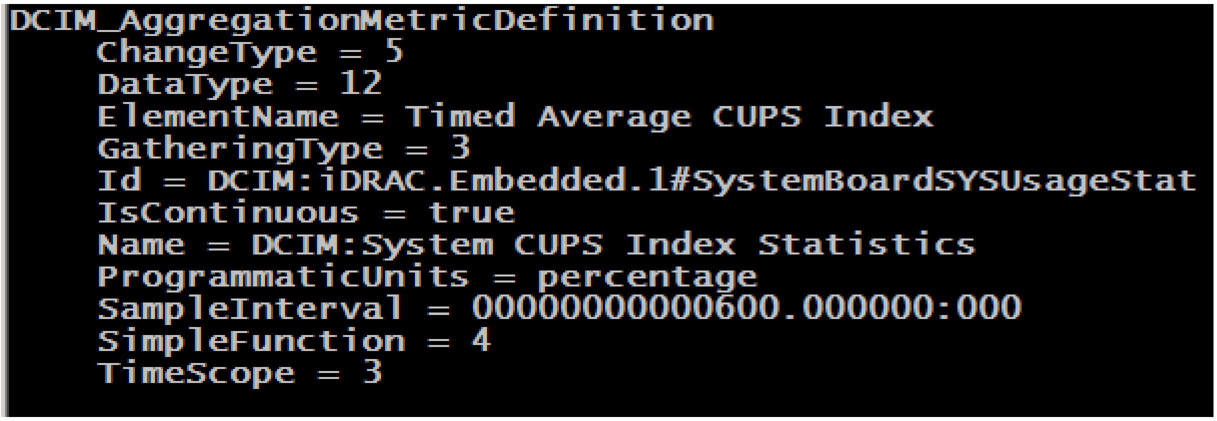
Winrm command:

```
winrm get http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/root/dcim/DCIM_AggregationMetricDefinition?__cimnamespace=root/dcim+Id=DCIM:<FQD
D> -u:<username> -p:<password> -r:https://<Ip-Address>/wsman -SkipCNcheck -SkipCAcheck -
SkipRevocationCheck -encoding:utf-8 -a:basic
```

Note:

- FQDD → Given in Table 2
- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
DCIM_AggregationMetricDefinition
    ChangeType = 5
    DataType = 12
    ElementName = Timed Average CUPS Index
    GatheringType = 3
    Id = DCIM:iDRAC.Embedded.1#SystemBoardSYSUsageStat
    IsContinuous = true
    Name = DCIM:System CUPS Index Statistics
    ProgrammaticUnits = percentage
    SampleInterval = 00000000000600.000000:000
    SimpleFunction = 4
    TimeScope = 3
```

Customers can use the programmatic interface of WS-Man to create their own management applications. So SOAP request packet for this class enumeration is given below:

SOAP XML :

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/transfer/Get</wsa:Action>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_AggregationMetricDefinition</wsman:ResourceURI>
...
<wsman:SelectorSet><wsman:Selector Name="__cimnamespace">root/dcim</wsman:Selector>
<wsman:Selector Name="Id">DCIM:FQDD</wsman:Selector>
</wsman:SelectorSet>
</s:Header>
<s:Body/>
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443
- FQDD → select the required one from the Table 2

2.2.1.3 Enumerate DCIM_AggregationMetricValue

DCIM_AggregationMetricValue is a derivation of CIM_AggregationMetricValue class.

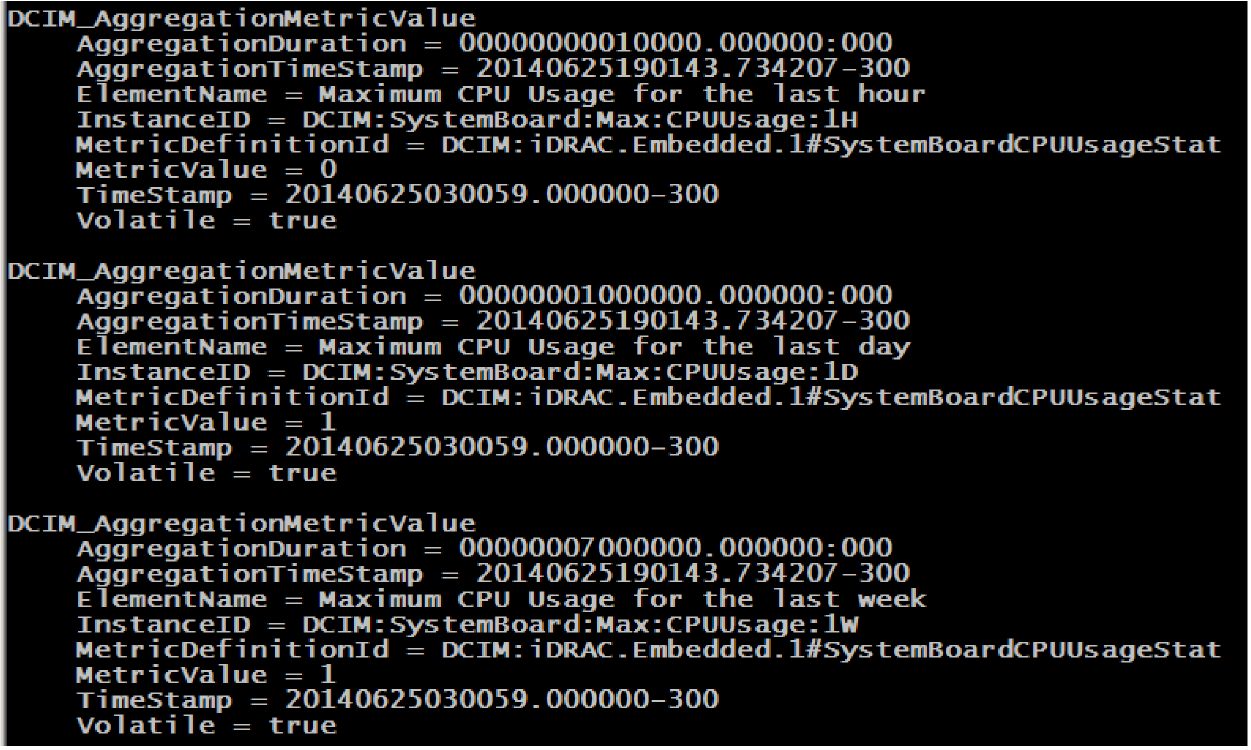
Winrm command:

```
winrm enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/root/dcim/DCIM_AggregationMetricValue -u:<username> -p:<password> -r:https://<Ip-
Address>/wsman -SkipCNcheck -SkipCAcheck -SkipRevocationCheck -encoding:utf-8 -a:basic
```

Note:

- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
DCIM_AggregationMetricValue
    AggregationDuration = 00000000010000.000000:000
    AggregationTimeStamp = 20140625190143.734207-300
    ElementName = Maximum CPU Usage for the last hour
    InstanceID = DCIM:SystemBoard:Max:CPUUsage:1H
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 0
    TimeStamp = 20140625030059.000000-300
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000001000000.000000:000
    AggregationTimeStamp = 20140625190143.734207-300
    ElementName = Maximum CPU Usage for the last day
    InstanceID = DCIM:SystemBoard:Max:CPUUsage:1D
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 1
    TimeStamp = 20140625030059.000000-300
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000007000000.000000:000
    AggregationTimeStamp = 20140625190143.734207-300
    ElementName = Maximum CPU Usage for the last week
    InstanceID = DCIM:SystemBoard:Max:CPUUsage:1W
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 1
    TimeStamp = 20140625030059.000000-300
    Volatile = true
```

Customers can use the programmatic interface of WS-Man to create their own management applications. So SOAP request packet for this class enumeration is given below:

SOAP XML:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/enumeration/Enumerate</wsa:Action>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-schema/2/DCIM_AggregationMetricValue</wsman:ResourceURI>
...
</wsman:SelectorSet>
</s:Header>
<s:Body>
<wsen:Enumerate>
<wsman:OptimizeEnumeration/>
<wsman:MaxElements>32000</wsman:MaxElements>
</wsen:Enumerate>
</s:Body>
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443

2.2.1.4 Get Instance of DCIM_AggregateMetricValue

Using get operation only desired instance can be viewed.The command is shown below:

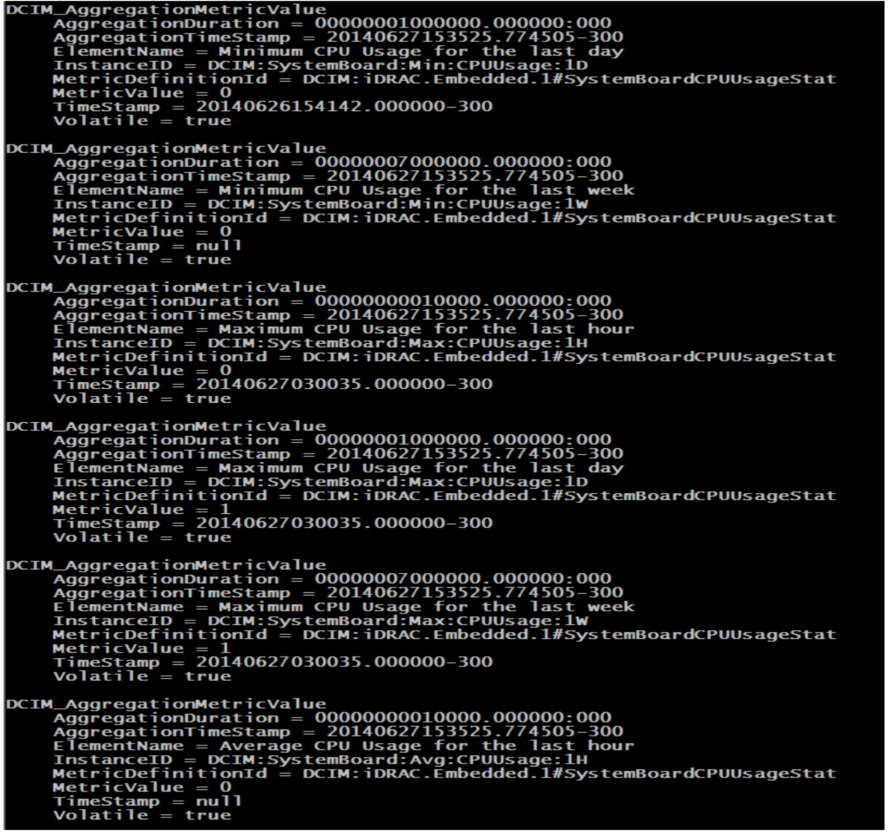
Winrm command:

```
winrm get http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/DCIM_AggregationMetricValue?__cimnamespace=root/dcim+InstanceID=<InstanceID> -u:<username> -p:<password> -r:https://<Ip-Address>/wsman -SkipCNcheck -SkipCAcheck -SkipRevocationCheck -encoding:utf-8 -a:basic
```

Note:

- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
DCIM_AggregationMetricValue
    AggregationDuration = 00000001000000.000000:000
    AggregationTimeStamp = 20140627153525.774505-300
    ElementName = Minimum CPU Usage for the last day
    InstanceID = DCIM:SystemBoard:Min:CPUUsage:1D
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 0
    TimeStamp = 20140626154142.000000-300
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000007000000.000000:000
    AggregationTimeStamp = 20140627153525.774505-300
    ElementName = Minimum CPU Usage for the last week
    InstanceID = DCIM:SystemBoard:Min:CPUUsage:1W
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 0
    TimeStamp = null
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000000010000.000000:000
    AggregationTimeStamp = 20140627153525.774505-300
    ElementName = Maximum CPU Usage for the last hour
    InstanceID = DCIM:SystemBoard:Max:CPUUsage:1H
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 0
    TimeStamp = 20140627030035.000000-300
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000001000000.000000:000
    AggregationTimeStamp = 20140627153525.774505-300
    ElementName = Maximum CPU Usage for the last day
    InstanceID = DCIM:SystemBoard:Max:CPUUsage:1D
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 1
    TimeStamp = 20140627030035.000000-300
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000007000000.000000:000
    AggregationTimeStamp = 20140627153525.774505-300
    ElementName = Maximum CPU Usage for the last week
    InstanceID = DCIM:SystemBoard:Max:CPUUsage:1W
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 1
    TimeStamp = 20140627030035.000000-300
    Volatile = true

DCIM_AggregationMetricValue
    AggregationDuration = 00000000010000.000000:000
    AggregationTimeStamp = 20140627153525.774505-300
    ElementName = Average CPU Usage for the last hour
    InstanceID = DCIM:SystemBoard:Avg:CPUUsage:1H
    MetricDefinitionId = DCIM:iDRAC.Embedded.1#SystemBoardCPUUsageStat
    MetricValue = 0
    TimeStamp = null
    Volatile = true
```

SOAP XML:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/transfer/Get</wsa:Action>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_AggregationMetricValue</wsman:ResourceURI>
...
<wsman:SelectorSet><wsman:Selector Name="__cimnamespace">root/dcim</wsman:Selector>
<wsman:Selector Name="InstanceID">DCIM:SystemBoard:Avg:CPUUsage:1D</wsman:Selector>
</wsman:SelectorSet>
</s:Header>
<s:Body/>
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443

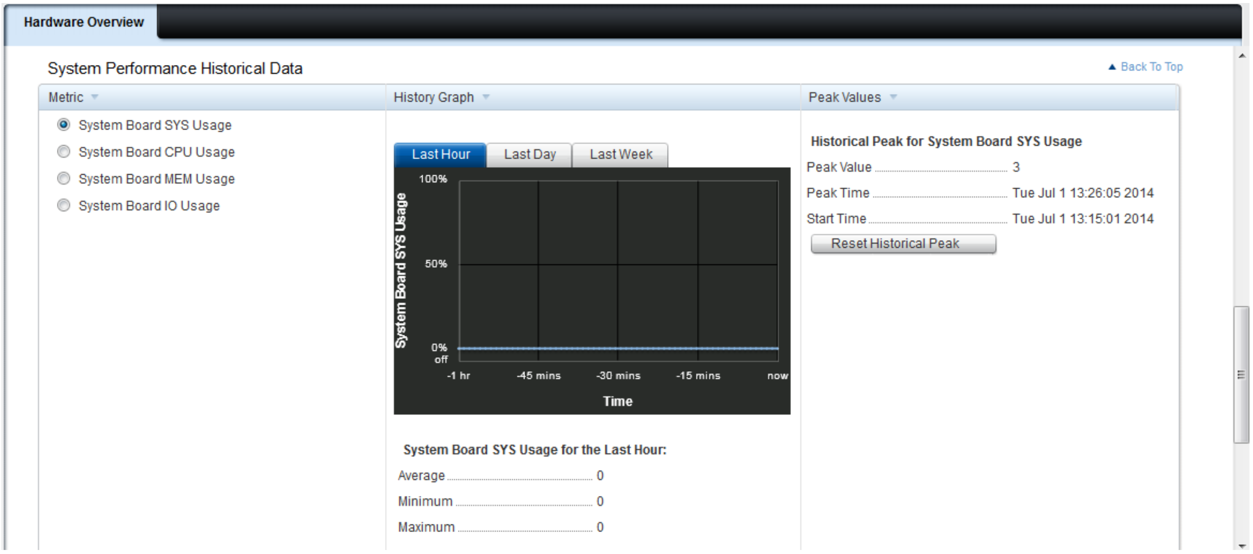
2.3 Using the iDRAC GUI

To view the statistical information of the Utilization monitoring sensors, traverse to the **System Performance Historical Data** table on **Hardware Overview** page.

The statistical information for last hour, last day and last week are presented in a graphical representation.

The average, minimum and maximum usage values are also displayed for last hour, last day and last week time periods.

The below snapshot provides the statistical information of the Utilization monitoring sensors



Hardware Overview

System Performance Historical Data

▲ Back To Top

Metric

● System Board SYS Usage
○ System Board CPU Usage
○ System Board MEM Usage
○ System Board IO Usage

History Graph

System Board SYS Usage for the Last Hour:

Average	0
Minimum	0
Maximum	0

Peak Values

Historical Peak for System Board SYS Usage

Peak Value	3
Peak Time	Tue Jul 1 13:26:05 2014
Start Time	Tue Jul 1 13:15:01 2014

Reset Historical Peak

3 Peak Reset

The utilization peak values can be reset from RACADM, WSMAN and WebGUI using the FQDD's defined in Table 2 for viewing the usage statistics.

3.1 Using RACADM

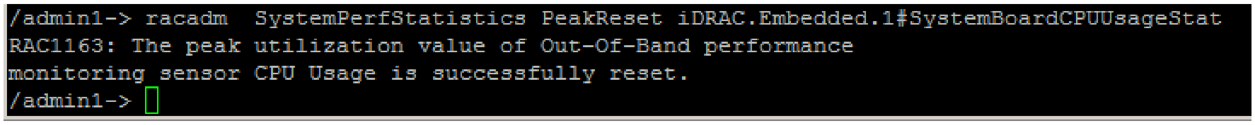
To reset the utilization peak values for the Utilization Monitoring sensors

```
racadm SystemPerfStatistics PeakReset <Statistics_Fqdd>
```

3.1.1 Usage Examples

To reset the utilization peak values for CPU sensor

```
racadm SystemPerfStatistics PeakReset iDRAC.Embedded.1#SystemBoardCPUUsageStat
```



```
/admin1-> racadm  SystemPerfStatistics PeakReset iDRAC.Embedded.1#SystemBoardCPUUsageStat
RAC1163: The peak utilization value of Out-Of-Band performance
monitoring sensor CPU Usage is successfully reset.
/admin1->
```

3.2 Using WS-Man

This section covers PeakReset usage examples.PeakReset is a method to reset the peak values captured for CPU, I/O, Memory and System Utilization

3.2.1 Usage Examples

This section explains WS-Man commands for resetting the peak value of Utilization Statistics:

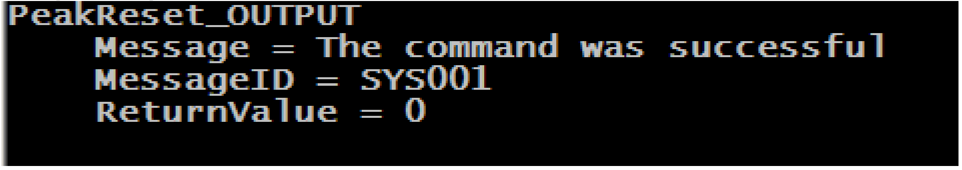
Winrm command:

```
winrm invoke PeakReset
cimv2/root/dcim/DCIM_MetricService?SystemCreationClassName=DCIM_ComputerSystem+CreationClassName=DCIM_MetricService+SystemName=DCIM:ComputerSystem+Name=DCIM:MetricService -u:<username> -p:<password> -r:https://<Ip-Address>/wsman -SkipCNcheck -SkipCAcheck -SkipRevocationCheck -encoding:utf-8 -a:basic @{UsageType="<FQDD>"}
```

Note:

- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
PeakReset_OUTPUT
    Message = The command was successful
    MessageID = SYS001
    ReturnValue = 0
```

Customers can use the programmatic interface of WS-Man to create their own management applications:
SOAP XML:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_MetricService/PeakReset</wsa:Action>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_MetricService</wsman:ResourceURI>
...
<wsman:OperationTimeout>PT300.0S</wsman:OperationTimeout>
<wsa:ReplyTo>
<wsa:Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:Address>
</wsa:ReplyTo>
<wsman:SelectorSet><wsman:Selector Name="__cimnamespace">root/dcim</wsman:Selector>
<wsman:Selector Name="SystemCreationClassName">DCIM_SPComputerSystem</wsman:Selector>
<wsman:Selector Name="SystemName">systemmc</wsman:Selector>
<wsman:Selector Name="CreationClassName">DCIM_MetricService</wsman:Selector>
<wsman:Selector Name="Name">DCIM MetricService 1</wsman:Selector></wsman:SelectorSet>
</s:Header>
<s:Body><n1:PeakReset_INPUT><n1:UsageType>FQDD</n1:UsageType></n1:PeakReset_INPUT></s:Bod
y>
</s:Envelope>
```

Note:

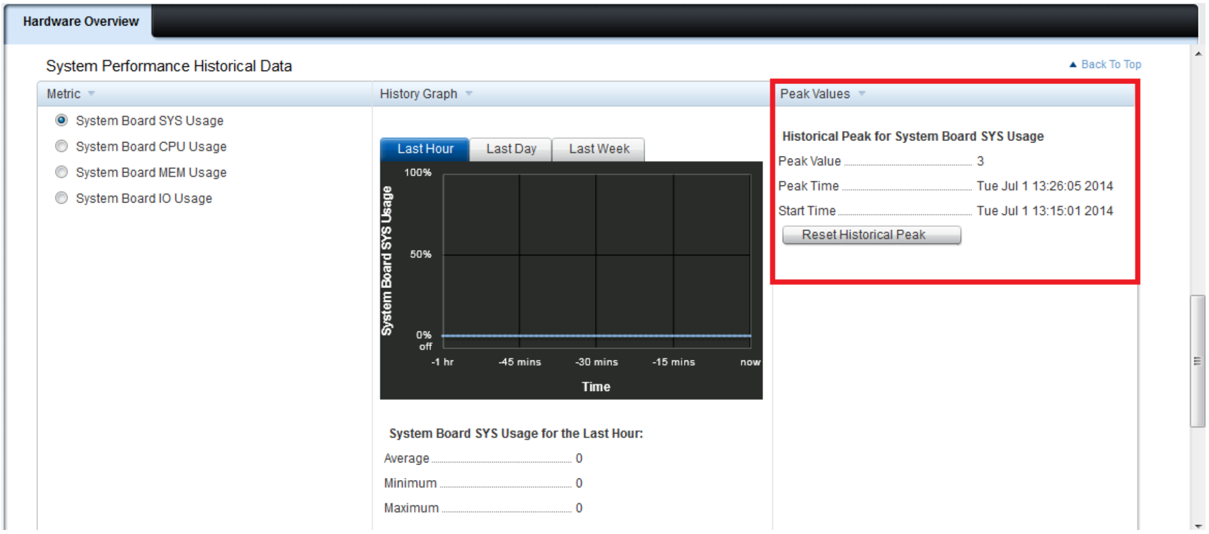
- IP:PORT → iDRAC IP Address:443
- FQDD → select the required one from the Table 2

3.3 Using the iDRAC GUI

To view and reset the utilization peak values for Utilization monitoring sensors traverse to **Peak values** table under **System Performance Historical Data** table on the **Hardware Overview** page.

Selcet the corresponding sensor to view the readings of Peak Value, Peak Time and Start Time.

“Peak Value” is the peak usage of the sensor, “Peak Time” is the time of the peak usage and “Start Time” is the time of the last reset operation.



Historical Peak for System Board SYS Usage

Peak Value 3
Peak Time .... Tue Jul 1 13:26:05 2014
Start Time .... Tue Jul 1 13:15:01 2014

System Board SYS Usage for the Last Hour:

Average 0
Minimum 0
Maximum 0

Click “**Reset Historical Peak**” button to reset the peak values of the sensors. This operation will reset the “Peak Value”, and update the “Peak Time” and “Start Time” of the corresponding sensor selected.

4 Sensor Threshold Values

The upper non-critical thresholds of the Utilization sensors can be viewed and configured by WS-Man and RACADM interfaces.

The upper non-critical threshold values of these sensors can be configured using the below FQDD's.

Table 2 FQDD of Utilization Sensors

Sensor Description	FQDD
CPU Utilization	iDRAC.Embedded.1#SystemBoardCPUUsage
IO Utilization	iDRAC.Embedded.1#SystemBoardIOUsage
Memory Utilization	iDRAC.Embedded.1#SystemBoardMEMUsage
System Utilization	iDRAC.Embedded.1#SystemBoardSYSUsage

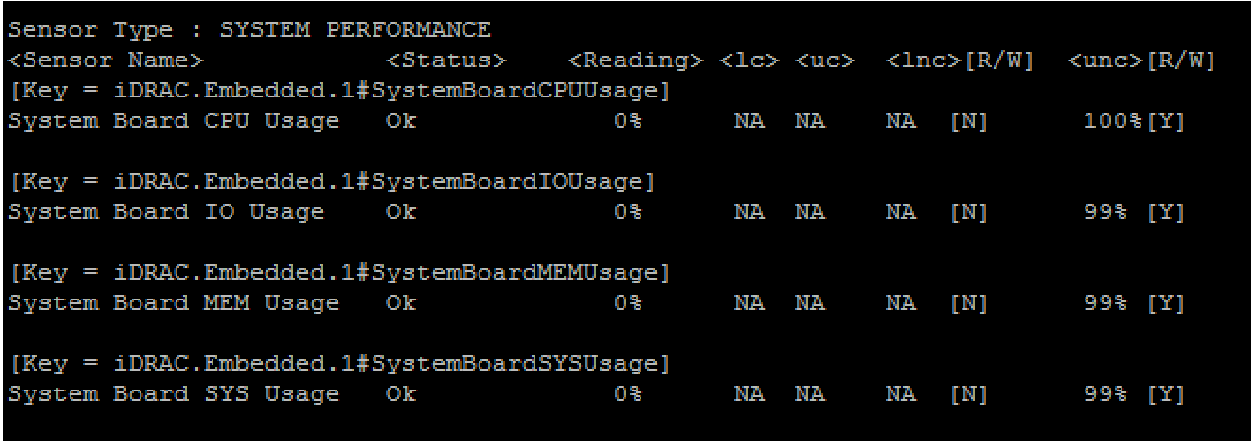
4.1 Using RACADM

To view the Utilization Monitoring sensor threshold values and its readings by running a RACADM command at Command Line Interface (CLI)

```
racadm getsensorinfo
```

This command provides the status of all the system sensors. It displays the Utilization sensor readings under the sensor type "SYSTEM PERFORMANCE". The command output also provides the sensor FQDD's of CPU, IO, Memory and System Utilization, using which the current thresholds of the sensors can be modified.

Sample Output:



```
Sensor Type : SYSTEM PERFORMANCE
<Sensor Name>            <Status>    <Reading> <lc> <uc>  <lnc>[R/W]  <unc>[R/W]
[Key = iDRAC.Embedded.1#SystemBoardCPUUsage]
System Board CPU Usage   Ok             0%      NA  NA    NA  [N]      100%[Y]

[Key = iDRAC.Embedded.1#SystemBoardIOUsage]
System Board IO Usage    Ok             0%      NA  NA    NA  [N]      99% [Y]

[Key = iDRAC.Embedded.1#SystemBoardMEMUsage]
System Board MEM Usage   Ok             0%      NA  NA    NA  [N]      99% [Y]

[Key = iDRAC.Embedded.1#SystemBoardSYSUsage]
System Board SYS Usage   Ok             0%      NA  NA    NA  [N]      99% [Y]
```

Note: In the above output:

- lc → Indicates the lower critical values of the sensor.
- lnc → Indicates the and non-critical threshold values of the sensor.
- uc → Indicates the upper critical values of the sensor.
- unc → Indicates the upper non-critical threshold values of the sensor.
- R/W (Read/Write) → Indicates whether the sensor thresholds are configurable.

To modify the current threshold settings of the sensor by running the command at CLI:

```
racadm sensorsettings set <Sensor_FQDD> -level <level> <value>
```

Note:

- level → "Min" for setting the lower thresholds and "Max" for setting the upper thresholds.
- Sensor_FQDD → Replace with FQDD mentioned in Table 1 for sensor type.

4.1.1 Usage Examples

To configure the upper non-critical threshold for the CPU sensor

```
racadm sensorsettings set iDRAC.Embedded.1#SystemBoardCPUUsage -level Max 95
```

Note:For remote RACADM, also enter the `-r <HostName> -u <user name> and -p <password>` options in the command.

4.2 Using WS-Man

DCIM_NumericSensor is the Dell derivation of the CIM_NumericSensor Class. DCIM_NumericSensor class represents non power related analog sensor. So Utilization sensors are implemented as per Numeric Sensors profile.The detailed analysis of this profile was given in Dell Sensors Profile and DMTF profile (DSP1009)documents.

- Utilization sensors can be viewed by enumerating the above class. A specific instance of an this sensor can be viewed using the Get Operation.
- The SettableThresholds property of this class contains thresholds that can be configured by the user for a particular sensor.

4.2.1 Usage Examples

This section explains WS-Man commands with examples for enumeration, Get and Set of Utilization Sensors. The meaning of values present in the output for each property is explained in Dell Sensors Profile document.

4.2.1.1 Enumerate DCIM_NumericSensor

The list of all Utilization sensors can be viewed using enumerate command given below:

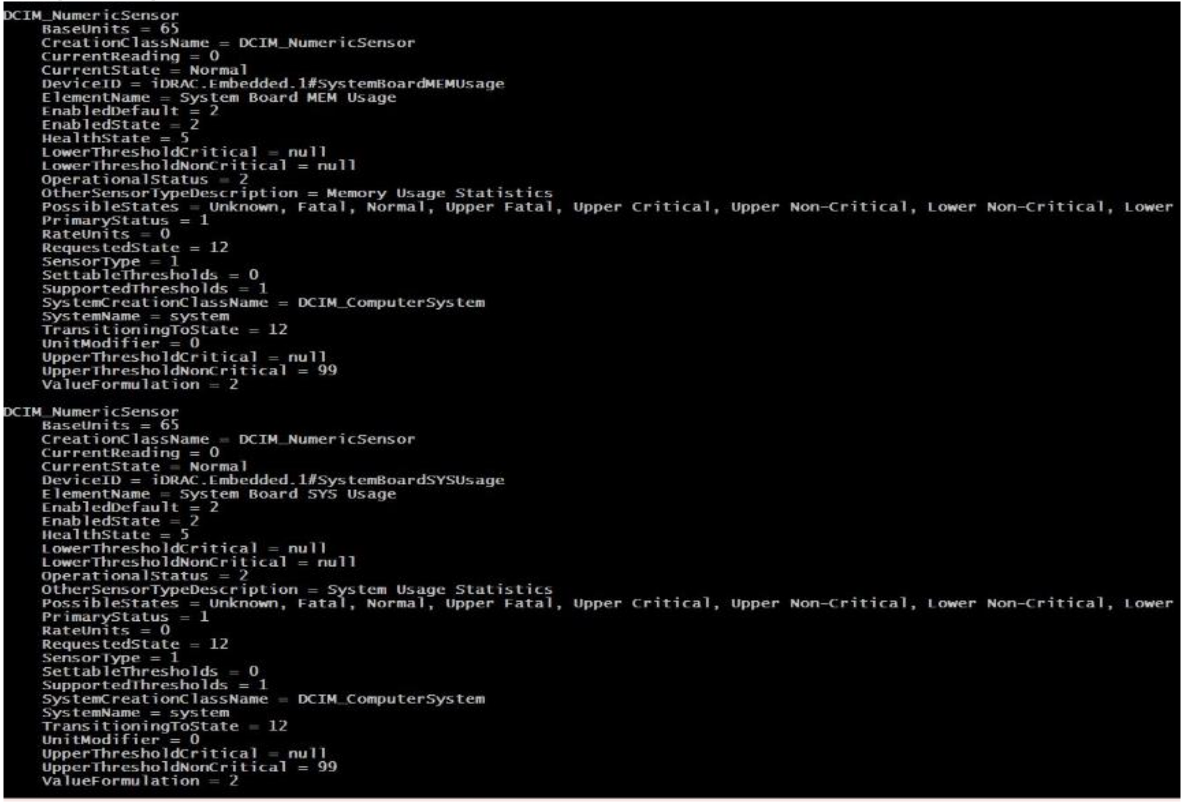
Winrm Command:

```
winrm enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/root/dcim/DCIM_NumericSensor -u:<username> -p:<password> -r:https://
<Ip-Address>/wsman -SkipCNcheck -SkipCAcheck -SkipRevocationCheck -encoding:utf-
8 -a:basic
```

Note:

- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
DCIM_NumericSensor
    BaseUnits = 65
    CreationClassName = DCIM_NumericSensor
    CurrentReading = 0
    CurrentState = Normal
    DeviceID = iDRAC.Embedded.1#SystemBoardMEMUsage
    ElementName = System Board MEM Usage
    EnabledDefault = 2
    EnabledState = 2
    HealthState = 5
    LowerThresholdCritical = null
    LowerThresholdNonCritical = null
    OperationalStatus = 2
    OtherSensorTypeDescription = Memory Usage Statistics
    PossibleStates = Unknown, Fatal, Normal, Upper Fatal, Upper Critical, Upper Non-Critical, Lower Non-Critical, Lower
    PrimaryStatus = 1
    RateUnits = 0
    RequestedState = 12
    SensorType = 1
    SettableThresholds = 0
    SupportedThresholds = 1
    SystemCreationClassName = DCIM_ComputerSystem
    SystemName = system
    TransitioningToState = 12
    UnitModifier = 0
    UpperThresholdCritical = null
    UpperThresholdNonCritical = 99
    ValueFormulation = 2

DCIM_NumericSensor
    BaseUnits = 65
    CreationClassName = DCIM_NumericSensor
    CurrentReading = 0
    CurrentState = Normal
    DeviceID = iDRAC.Embedded.1#SystemBoardSYSUsage
    ElementName = System Board SYS Usage
    EnabledDefault = 2
    EnabledState = 2
    HealthState = 5
    LowerThresholdCritical = null
    LowerThresholdNonCritical = null
    OperationalStatus = 2
    OtherSensorTypeDescription = System Usage Statistics
    PossibleStates = Unknown, Fatal, Normal, Upper Fatal, Upper Critical, Upper Non-Critical, Lower Non-Critical, Lower
    PrimaryStatus = 1
    RateUnits = 0
    RequestedState = 12
    SensorType = 1
    SettableThresholds = 0
    SupportedThresholds = 1
    SystemCreationClassName = DCIM_ComputerSystem
    SystemName = system
    TransitioningToState = 12
    UnitModifier = 0
    UpperThresholdCritical = null
    UpperThresholdNonCritical = 99
    ValueFormulation = 2
```

Customers can use the programmatic interface of WS-Man to create their own management applications. So SOAP request packet for this class enumeration is given below:

SOAP XML:

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/enumeration/Enumer
ate</wsa:Action>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI
s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_NumericSensor</wsman:ResourceURI>
...
<wsman:SelectorSet>
...
</wsman:SelectorSet>
</s:Header>
...
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443

4.2.1.1.1 Get Instance of DCIM_NumericSensor

Using get operation, a particular Utilization Sensor can be viewed as shown below:

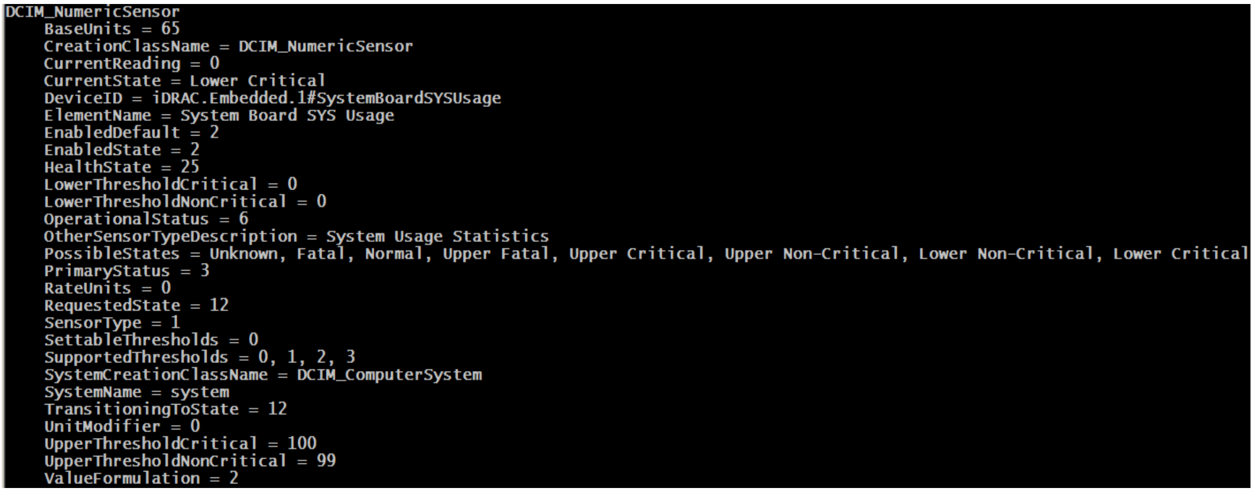
Winrm Command:

```
winrm get http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_NumericSensor?__cimnamespace=root/dcim+SystemCreationClassName=DCIM_Com
puterSystem+SystemName=system+CreationClassName=DCIM_NumericSensor+DeviceID=<FQDD> -
u:<username> -p:<password> -r:https://<Ip-Address>/wsman -SkipCNcheck -SkipCAcheck -
SkipRevocationCheck -encoding:utf-8 -a:basic
```

Note:

- FQDD → Given in Table 1
- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.

Sample Output:



```
DCIM_NumericSensor
    BaseUnits = 65
    CreationClassName = DCIM_NumericSensor
    CurrentReading = 0
    CurrentState = Lower Critical
    DeviceID = iDRAC.Embedded.1#SystemBoardSYSUsage
    ElementName = System Board SYS Usage
    EnabledDefault = 2
    EnabledState = 2
    HealthState = 25
    LowerThresholdCritical = 0
    LowerThresholdNonCritical = 0
    OperationalStatus = 6
    OtherSensorTypeDescription = System Usage Statistics
    PossibleStates = Unknown, Fatal, Normal, Upper Fatal, Upper Critical, Upper Non-Critical, Lower Non-Critical, Lower Critical
    PrimaryStatus = 3
    RateUnits = 0
    RequestedState = 12
    SensorType = 1
    SettableThresholds = 0
    SupportedThresholds = 0, 1, 2, 3
    SystemCreationClassName = DCIM_ComputerSystem
    SystemName = system
    TransitioningToState = 12
    UnitModifier = 0
    UpperThresholdCritical = 100
    UpperThresholdNonCritical = 99
    ValueFormulation = 2
```

Customers can use the programmatic interface of WS-Man to create their own management applications. They can use the below SOAP XML

SOAP XML

```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:Action
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/transfer/Get</wsa:Action>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/DCIM_NumericSensor</wsman:ResourceURI>
…
<wsman:SelectorSet><wsman:Selector Name="__cimnamespace">root/dcim</wsman:Selector>
<wsman:Selector Name="DeviceID">FQDD</wsman:Selector>
<wsman:Selector Name="CreationClassName">DCIM_NumericSensor</wsman:Selector>
<wsman:Selector Name="SystemName">system</wsman:Selector>
<wsman:Selector Name="SystemCreationClassName">DCIM_ComputerSystem</wsman:Selector>
</wsman:SelectorSet>
</s:Header>
<s:Body/>
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443
- FQDD → Given in Table 1

4.2.1.2 Set Upper Threshold Value

Using set operation, Upper Non-Critical threshold values can be configured.The commands are given below:

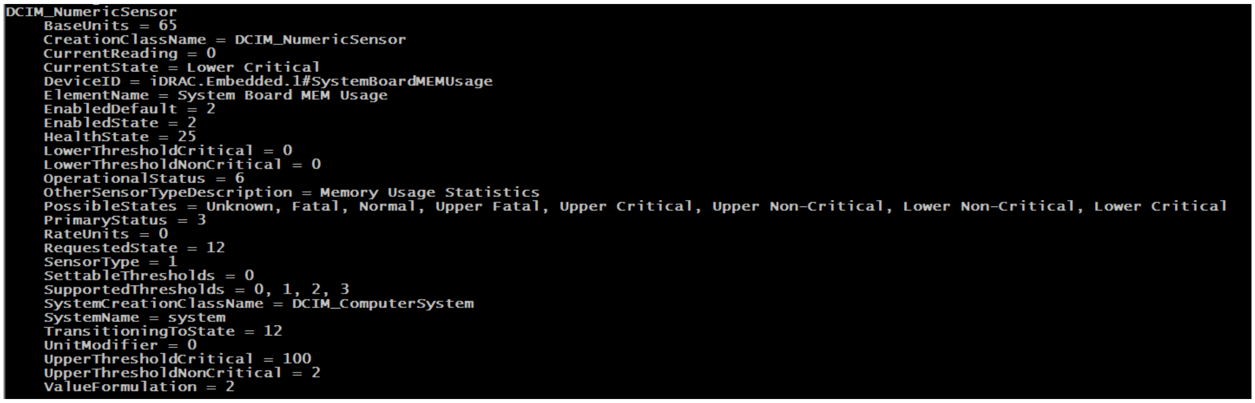
Winrm Command:

```
winrm set "http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/DCIM_NumericSensor?DeviceID=<FQDD>+CreationClassName=DCIM_NumericSensor+Syste
mCreationClassName=DCIM_ComputerSystem+SystemName=system" -u:<Username> -p:<Password> -
r:https://Ip-Address/wsman -SkipCNcheck -SkipCAcheck -SkipRevocationCheck -encoding:utf-8 -
a:basic @{UpperThresholdNonCritical="<value>"}
```

Note:

- FQDD → Given in Table 1
- Username → iDRAC login username
- Password → iDRAC login password
- Ip-Address → iDRAC IP Address.
- Value → value to which parameter is set.

Sample Output:



```
DCIM_NumericSensor
    BaseUnits = 65
    CreationClassName = DCIM_NumericSensor
    CurrentReading = 0
    CurrentState = Lower Critical
    DeviceID = iDRAC.Embedded.1#SystemBoardMEMUsage
    ElementName = System Board MEM Usage
    EnabledDefault = 2
    EnabledState = 2
    HealthState = 25
    LowerThresholdCritical = 0
    LowerThresholdNonCritical = 0
    OperationalStatus = 6
    OtherSensorTypeDescription = Memory Usage Statistics
    PossibleStates = Unknown, Fatal, Normal, Upper Fatal, Upper Critical, Upper Non-Critical, Lower Non-Critical, Lower Critical
    PrimaryStatus = 3
    RateUnits = 0
    RequestedState = 12
    SensorType = 1
    SettableThresholds = 0
    SupportedThresholds = 0, 1, 2, 3
    SystemCreationClassName = DCIM_ComputerSystem
    SystemName = system
    TransitioningToState = 12
    UnitModifier = 0
    UpperThresholdCritical = 100
    UpperThresholdNonCritical = 2
    ValueFormulation = 2
```

Customers can use the programmatic interface of WS-Man to create their own management applications:

SOAP XML

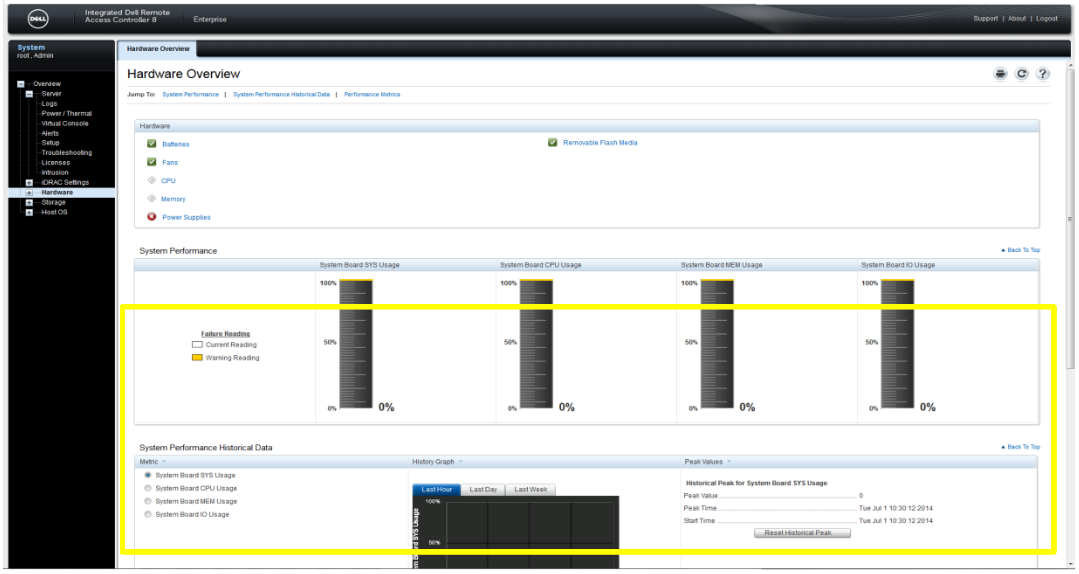
```
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope ...>
<s:Header>
<wsa:To s:mustUnderstand="true">https://IP:PORT/wsman</wsa:To>
<wsman:ResourceURI s:mustUnderstand="true">http://schemas.dmtf.org/wbem/wscim/1/cim-
schema/2/DCIM_NumericSensor</wsman:ResourceURI>
<wsa:ReplyTo><wsa:Address
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa:A
ddress></wsa:ReplyTo>
<wsa:Action
s:mustUnderstand="true">http://schemas.xmlsoap.org/ws/2004/09/transfer/Put</wsa:Action>
...
<wsman:SelectorSet><wsman:Selector Name="DeviceID">FQDD</wsman:Selector>
<wsman:Selector Name="CreationClassName">DCIM_NumericSensor</wsman:Selector>
<wsman:Selector Name="SystemName">system</wsman:Selector>
<wsman:Selector Name="SystemCreationClassName">DCIM_ComputerSystem</wsman:Selector>
</wsman:SelectorSet>
...
</s:Header>
<s:Body><n1:DCIM_NumericSensor>
...
<n1:UpperThresholdNonCritical>VALUE</n1:UpperThresholdNonCritical>
...</n1:DCIM_NumericSensor></s:Body>
</s:Envelope>
```

Note:

- IP:PORT → iDRAC IP Address:443
- FQDD → Given in Table 1
- VALUE → value to which parameter is set.

4.3 Using the iDRAC GUI

- Log in to the iDRAC Web GUI by typing the IP address or hostname of the iDRAC in the address bar, and then typing appropriate information.
- In the left plane, click **Hardware** to view the Utilization sensors.



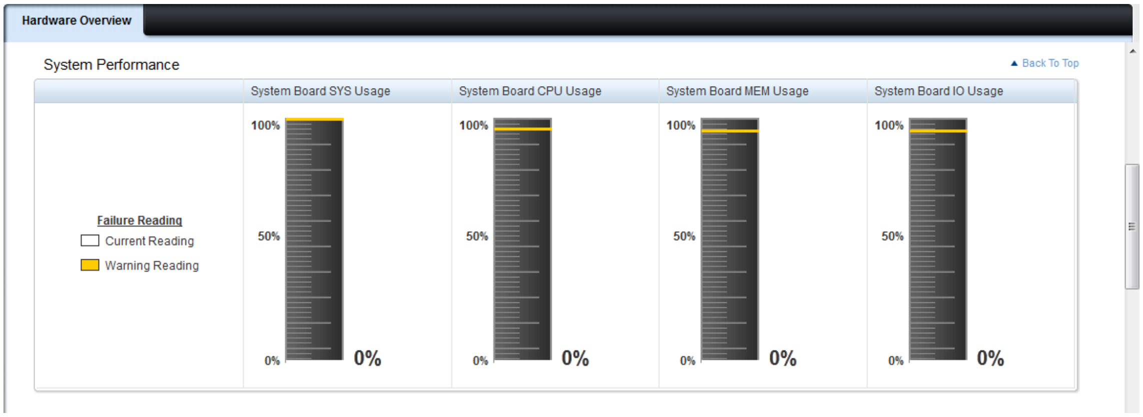
Historical Peak for System Board SYS Usage

Peak Value 0

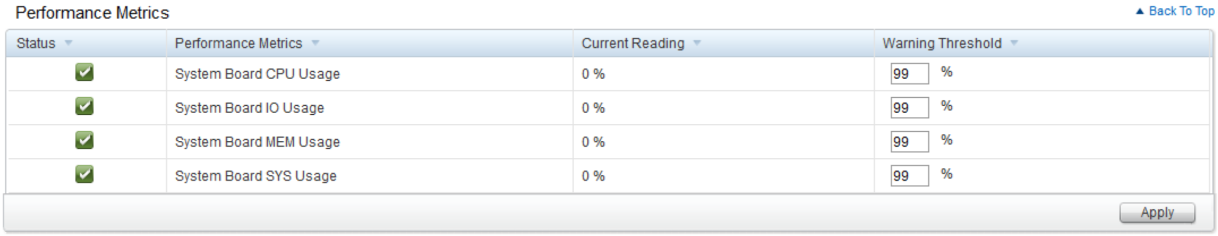
Peak Time .... Tue Jul 1 10:30:12 2014

Start Time .... Tue Jul 1 10:30:12 2014

The **System Performance** table on **Hardware Overview** page, provides the current readings and warning threshold (upper non-critical) values for System, CPU, Memory and IO sensors.



Traverse to **Performance Metrics** table on **Hardware Overview** page to configure the warning (upper non-critical) threshold values. User needs to enter the threshold values in the text box corresponding to the sensor and then click on “**Apply**” to configure the values.



Performance Metrics

Status	Performance Metrics	Current Reading	Warning Threshold
✓	System Board CPU Usage	0 %	99 %
✓	System Board IO Usage	0 %	99 %
✓	System Board MEM Usage	0 %	99 %
✓	System Board SYS Usage	0 %	99 %

5 Events/Alert Information

The IPMI SEL, SNMP traps are getting generated if the utilization index crosses the upper warning threshold. By default these sensor events/alerts are not enabled. It can be enabled by standard IPMI command for each individual sensor. As per IPMI specifications 2.0 (Link: http://www.intel.in/content/www/in/en/servers/ipmi/ipmi-second-gen-interface-spec-v2-rev1-1.html), sensor event generation can be enabled using "Set Sensor Event Enable Command" (section 35.10 of IPMI specifications 2.0).

These commands are as follows:

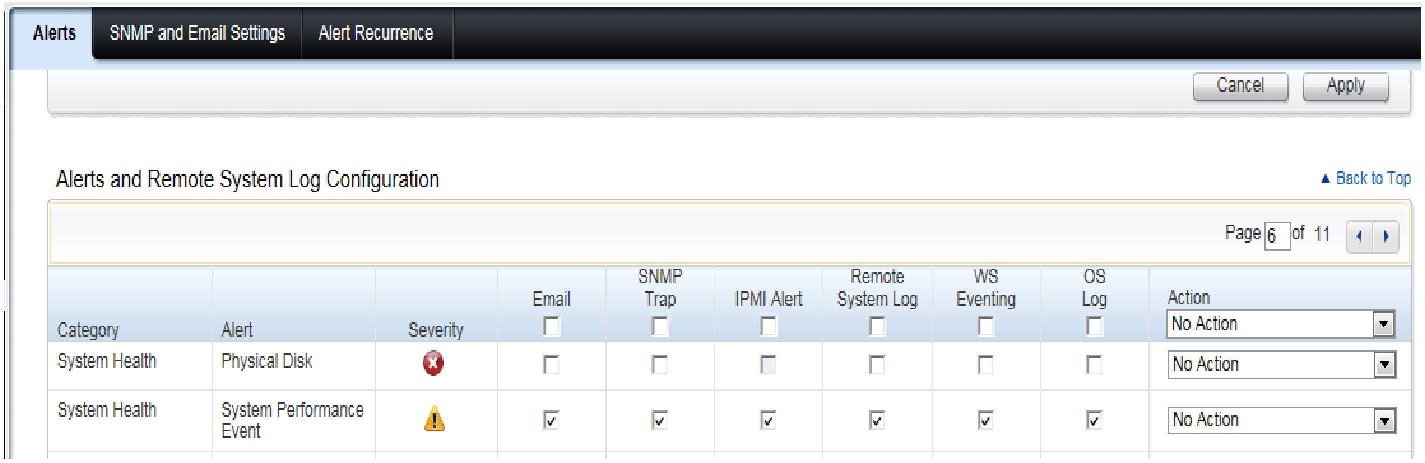
1. CPU Usage sensor events:
   NetFun: 0x04
   Command ID: 0x28
   1st Data Byte : 0xF0 (Sensor Num for CPU Usage sensor)
   (Note: Starting from iDRAC 2.05.05.00, this CPU Usage sensor is changed to 0xFD)
   Rest of data bytes: 0xC0, 0x80, 0x00, 0x80, 0x00
2. IO Usage sensor events: replace the 1st Data byte with 0xF1 instead of 0xF0
3. Mem Usage sensor events: 1st Data byte to be 0xF2
4. System Level Usage sensor events: 1st Data byte to be 0xF3

Here is example of raw IPMI commands for all these sensor if local IPMITool is used:

1. CPU: IPMITool raw 0x04 0x28 0xF0 0xC0 0x80 0x00 0x80 0x00 (iDRAC 2.00.00.00 release only)
   : IPMITool raw 0x04 0x28 0xFD 0xC0 0x80 0x00 0x80 0x00 (iDRAC 2.05.05.00 and all other future releases)
2. IO: IPMITool raw 0x04 0x28 0xF1 0xC0 0x80 0x00 0x80 0x00
3. Mem: IPMITool raw 0x04 0x28 0xF2 0xC0 0x80 0x00 0x80 0x00
4. System Level: IPMITool raw 0x04 0x28 0xF3 0xC0 0x80 0x00 0x80 0x00

Once the sensor events are enabled, IPMI SEL will be logged if the utilization index reaches to warning threshold index. The default warning threshold is set to 99%.

The SNMP alerts, IPMI Alerts can be enabled from Alerts page on iDRAC GUI.



Alerts and Remote System Log Configuration

Page 6 of 11

Category	Alert	Severity	Email □	SNMP Trap □	IPMI Alert □	Remote System Log □	WS Eventing □	OS Log □	Action No Action
System Health	Physical Disk	⊗	□	□	□	□	□	□	No Action
System Health	System Performance Event	⚠	☑	☑	☑	☑	☑	☑	No Action

6 How do we use this performance indexes?

The performance data provided by iDRAC is different than what other performance tools like SAR on linux , PERFMON on windows provide. These OS based tools provide measure of amount of resource utilization, whereas iDRAC provides measure of bandwidth utilization of a particular resource (CPU/IO/Mem). These indexes help in understanding nature of workload. E.g. If the Memory index is higher than other indexes, that means the workloads running on the server are memory intensive.

The following table shows comparison matrix:

CUPS & OS-Based Utilization Reports(SAR = Linux System Activity Report; PERFMON = Windows Performance Monitor)

	CUPS measures:	**SAR & PERFMON measure:**
CPU Utilization	% of CPU resources in an active state (C0)	% of available physical& logical cores allocatedto threads, regardless of activity (C0 or Sx)
Memory Utilization	% of total Memorytransfer bandwidth in active use	% of total system Memory capacity allocated to scheduled threads
I/O Utilization	% of total PCIe bandwidthin active use	% of total disk & network packet transfers reported at respective devices
Aggregate Utilization per Time Interval	%of aggregated CPU, Memory & I/O resources in active use	No comparable metric

Which is Better, CUPS or OS-Based Metrics?

- Appropriate metric depends on datacenter administrator’s goals & constraints

OS-Based Tools may be more appropriate when:

- Minimizing potential resource conflicts is the priority when placing workloads
- Workload placement does not need to consider power consumption
- Planning server configurations for purchase (e.g. total memory configuration)
- In-band data retrieval is required

CUPS metric may be more appropriate when:

- Workload placement is based on actual resource activity and power budget
- Tolerance for potential resource conflicts in SLA is higher
- Out-of-band data retrieval is required

A Appendix

Table 3 Component table example

Component	Description
Operating system	OS type, patch level, service pack, hypervisor, etc.
Driver version	NIC driver, storage drivers, etc.
Firmware version	NIC, RAID, switch, storage firmware versions, etc.
Application	i.e. SQL Server, Oracle, etc. Includes management applications (OMSA, SANHQ), etc.
Cabling	If applicable – power, network, and/or other cabling
Server	Server type, CPU type, memory, internal disks, etc.
Storage	Model, disk type and #,
Switch	Model, firmware/OS, etc.

A.1 References

Component	Description
DSP1009 Sensors Profile	http://www.dmtf.org/sites/default/files/standards/documents/DSP1009_1.0.2.pdf
DSP1053 Base Metrics Profile	http://www.dmtf.org/sites/default/files/standards/documents/DSP1053_1.0.1.pdf
Dell Base Metrics Profile 1.2	http://en.community.dell.com/techcenter/extras/m/white_papers/20439300.aspx
Dell Sensors Profile 1.1	http://en.community.dell.com/techcenter/extras/m/white_papers/20263530.aspx
Dell Message Registry – English(2.1)	http://en.community.dell.com/dell-groups/dtcmedia/m/mediagallery/20254139.aspx
Creating Management Application using WS-Man API	http://en.community.dell.com/techcenter/extras/m/white_papers/20439340.aspx

A.1.1 Acronyms

Acronym	
CLI	Command Line Interface
CPU	Central Processing Unit
CUPS	Compute Usage Per Second
FQDD	Fully Qualified Device Descriptor
GUI	Graphical User Interface
iDRAC	Integrated Dell Remote Access Controller
I/O	Input / Output
IPMI	Intelligent Platform Management Interface
ME	Management Engine
OOB	Out-Of Band
PCI	Peripheral Component Interconnect
RACADM	Remote Access Controller ADMinistration
RMC	Resource Monitoring Counter
RMON	Resource MONitor
SEL	System Event Log
SNMP	Simple Network Management Protocol
WS-Man	Web Services Management
SAR	System Activity Report
PERFMON	Windows Performance Monitor