

Cluster Monitoring: How to collect metrics from older GPUs using NVML

The Bright 8.0 metrics system uses DCGM to collect metrics from GPUs. But DCGM doesn't support older GPUs.

Not to worry, you can still get the metrics from them using NVML using the old metrics collection script, which is still installed by default on Bright 8.0 clusters.

```
/cm/local/apps/cmd/scripts/metrics/sample_gpu
```

1. Add a new data producer, which is the old NVML metrics collection script.

```
[root@virgo-head ~]# cmlsh
[virgo-head]% monitoring
[virgo-head->monitoring]% setup
[virgo-head->monitoring->setup]% add collection sample-nvml-gpu
[virgo-head->monitoring->setup*[sample-nvml-gpu*]]% set script /cm/local/apps/cmd/scripts/metrics/sample_gpu
```

2. Add a new node execution filter. The data producer will only run on the nodes that have this "NVML" resource defined.

```
[virgo-head->monitoring->setup*[sample-nvml-gpu*]]% nodeexecutionfilters
[virgo-head->monitoring->setup*[sample-nvml-gpu*]->nodeexecutionfilters*]% add resource NVML
[virgo-head->monitoring->setup*[sample-nvml-gpu*]->nodeexecutionfilters*[NVML*]]% set resources NVML
[virgo-head->monitoring->setup*[sample-nvml-gpu*]->nodeexecutionfilters*[NVML*]]% commit
```

3. Add a userdefinedresource to the GPU node(s) you want the data producer to run on.

```
[virgo-head->monitoring->setup[sample-nvml-gpu]->nodeexecutionfilters[NVML]]% device use gpu01
[virgo-head->device[gpu01]]% append userdefinedresources NVML
[virgo-head->device*[gpu01*]]% commit
```

Cluster Monitoring: How to collect metrics from older GPUs using NVML

4. Demonstrate that the metrics are now being collected (using NVML).

```
[virgo-head->device[gpu01]]% samplenow --metrics | grep gpu
Bar1MemFreeGPU          0          gpu          265 Mbytes
    0.232s
Bar1MemUsedGPU          0          gpu          2.62 Mbytes
    0.232s
DecoderUtilGPU          0          gpu          0.0%
    0.232s
EccDBitGPU              0          gpu          0 err
    0.232s
EccSBitGPU              0          gpu          0 err
    0.232s
EncoderUtilGPU          0          gpu          0.0%
    0.232s
FanSpeedPercGPU        0          gpu          2600.0%
    0.232s
GpuUtilGPU              0          gpu          0.0%
    0.232s
MemFreeGPU              0          gpu          11.9 Gbytes
    0.232s
MemUsedGPU              0          gpu          0 bytes
    0.232s
MemUtilGPU              0          gpu          0.0%
    0.232s
PcieReplayCounterGPU   0          gpu          0 replays
    0.232s
PowerDrawGPU            0          gpu          20.177 W
    0.232s
ProcsComputeGPU         0          gpu          0 processes
    0.232s
ProcsGraphicsGPU       0          gpu          0 processes
    0.232s
TempGPU                 0          gpu          36 C
    0.232s
```

Unique solution ID: #1381

Author: Robert Stober

Last update: 2017-08-26 16:29