nvidia-healthmon
Best Practices Guide

Version 2.3

NVIDIA Corporation

June 27, 2013

Contents

Chapter 1

Scope of this document

This document covers the use cases and best practices NVIDIA recommends for nvidia-healthmon . Let us know on the forums if your use case is not mentioned, https://devtalk.nvidia.com/. This document will not cover details of nvidia-healthmon use. This is covered in the nvidia-healthmon User Guide.

Chapter 2

Overview

nvidia-healthmon is the system administrator's and cluster manager's tool for detecting and troubleshooting common problems affecting NVIDIA®Tesla™GPUs in a high performance computing environment. nvidia-healthmon contains limited hardware diagnostic capabilities, and focuses on software and system configuration issues.

2.1 nvidia-healthmon goals

nvidia-healthmon is designed to:

1. Discover common problems that affect a GPUs ability to run a compute job including
   (a) Software configuration issues
   (b) System configuration issues
   (c) System assembly issues, like loose cables
   (d) A limited number of hardware issues
2. Provide troubleshooting help
3. Easily integrate into Cluster Scheduler and Cluster Management applications
4. Reduce downtime and failed GPU jobs

2.1.1 Beyond the scope of nvidia-healthmon

nvidia-healthmon is not designed to:

1. Provide comprehensive hardware diagnostics
2. Actively fix problems

Chapter 3

Use cases

While nvidia-healthmon is primarily targeted at clusters of NVIDIA®Tesla™GPUs, it can also be used in workstations without a cluster manager.

3.1 Run nvidia-healthmon after system provisioning

After a system is provisioned, nvidia-healthmon can be run on the node to ensure that the node is correctly configured and able to run a GPU job. In this use case an extended mode run of nvidia-healthmon will try to deliver the most comprehensive system health check.

3.2 Run nvidia-healthmon before a job

nvidia-healthmon can be run in a prologue or epilogue script in quick mode to perform a sanity check of the system and GPU. If nvidia-healthmon detects a problem the scheduler can mark the current node as down, and run the job on a different node to avoid job failure. In quick mode a subset of tests are run, such that a quick sanity test of the system is performed.

Note that nvidia-healthmon will create a CUDA context on the device it is testing so it is often undesirable to run nvidia-healthmon when other processes are using the GPU to prevent unexpected behavior of either process.

3.3 Periodic health check

Analogous to periodically scanning for viruses, nvidia-healthmon can be run in the extended mode periodically. Again, when nvidia-healthmon reports a problem the scheduler can mark the node as down.

3.4 After job failure

When a GPU job fails, the extended mode run of nvidia-healthmon can help troubleshoot the problem.

If the system is very unstable such that crashes or hangs are seen, consider running nvidia-healthmon with the '-i' or '–id' to individually target each GPU on the system. Use the PCI Bus ID or NVML index of the GPU for this. On multi-GPU systems such issues are often related to a single GPU, so nvidia-healthmon will run to completion on the working GPUs, but will fail on a single non-working GPU. Following this procedure can help isolate the issue to a single GPU, which helps narrow down potential problems.

3.5 Interfacing with other services

nvidia-healthmon can be run by a wrapper script which handles any reported warnings and errors. These problems can subsequently be forwarded to other services in order to notify them of any problems. NVIDIA suggests using `syslog` for logging reported issues on the system. Similarly, an SNMP trap can be sent to notify hosts over a network of these issues.

3.6 Troubleshooting problems

nvidia-healthmon 's troubleshooting report is designed to cover common problems, and will often suggest a number of possible solutions. These troubleshooting steps should be tackled from the top down, as the most likely solution is listed at the top.

3.6.1 Save log files

NVIDIA recommends that the log files from failing nvidia-healthmon runs should be saved. Saving log files ensures that data about intermittent problems is not lost.

Note: Some log files are encrypted and can only be decoded by NVIDIA engineers. These files are not corrupt. These logs contain a trace of the nvidia-healthmon run, and do not contain any sensitive information.

Chapter 4

Configuration

The config.ini sample that ships with nvidia-healthmon is configured with sone default settings for current generation GPUs. Since many of the fields in this file are sensitive to system configuration, NVIDIA provides very conservative estimates for various fields. Other fields are too system configuration dependent, so NVIDIA doesn't provide a default at all. To get the most out of nvidia-healthmon , NVIDIA recommends updating this config.ini file with measurements taken from the target system. One should use caution here, as setting values too tightly may cause nvidia-healthmon to report issues too frequently. A system architect or system administrator may be in the best position to decide what the optimal values should be.

Chapter 5

NUMA Systems

The nvidia-healthmon bandwidth test can give noticeably different results between runs on a NUMA system. To work around this issue, one can build a script that determines the CPU that is closest to each GPU in the system. lspci and lstopo can perform this task. The highest bandwidth will be seen between a GPU and the CPU it is closest to. numactl can be used to bind nvidia-healthmon to a particular CPU. For best results, run nvidia-healthmon multiple times. Each run, bind nvidia-healthmon to a different CPU, and target all GPUs that are closest to the bound CPU.

Note: Future versions of nvidia-healthmon may perform this binding automatically and dynamically such that only one run is needed, and all bandwidth tests use the fastest CPU-GPU pair.

Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication of otherwise under any patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all other information previously supplied. NVIDIA Corporation products are not authorized as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

Copyright

NVIDIA.