



Search NVIDIA

USA - United States

DRIVERS

PRODUCTS

DEEP LEARNING AND AI

COMMUNITIES

SUPPORT

SHOP

ABOUT NVIDIA

NVIDIA SUPPORT

NVIDIA Home > Support Home Page > Knowledgebase Home Page > [USING CUDA AND X](#)

[Log In](#) | [Sign Up](#)

USING CUDA AND X

Answer ID 3029

Updated 08/22/2013 03:36 PM

Using CUDA and X

When GPUs are used to simultaneously function as a General Computing Processor using the NVIDIA CUDA architecture and also as a display manager using X Windows, the user should be aware of certain limitations with handling both activities simultaneously on a single GPU.

If no consideration is given to managing both sets of tasks simultaneously, the system may experience disturbances and hangs in the X Window system, leading to an interruption of processing X-related tasks, such as display updates and rendering.

RECOMMENDATIONS FOR CUDA AND X

There are several options for managing the interactivity requirements of X while performing CUDA processing tasks.

Option 1: Use Two GPUs (RECOMMENDED)

If two GPUs can be made available in the system, then X processing can be handled on one GPU while CUDA tasks are executed on the other. This allows full interactivity and no disturbance of X while simultaneously allowing unhindered CUDA execution.

In order to accomplish this:

- The X display should be forced onto a single GPU using the BusID parameter in the relevant "Device" section of the xorg.conf file. In addition, any other "Device" sections should be deleted. For example:

```
BusID "PCI:34:0:0"
```

The PCI IDs of the GPUs may be determined from the lspci command or from the nvidia-smi -a command.

- CUDA processing should be forced onto the other GPU, for example by using the CUDA_VISIBLE_DEVICES environment variable before any CUDA applications are launched. For example:

```
export CUDA_VISIBLE_DEVICES="1" (Choose the numerical parameter to select the GPU that is not the X GPU)
```

Option 2: Disable X

If there is no display to be managed and no X rendering tasks (for example, OpenGL) required, then X can usually be disabled. This may be accomplished by deleting the X system from the system configuration, or by booting the system into runlevel 3 instead of runlevel 5.

Eliminating X (if not required) is generally desirable in cluster computing situations. For occasional use, the node can be started in runlevel 3 and then X can be started separately using xinit if needed.

Option 3: Limit CUDA Kernels to Short Execution Times

Generally speaking, if the maximum execution time of any given CUDA kernel is less than 0.1 seconds, the effect on the X Window system and interactivity should be minimal.

the X window system and interactivity should be minimal.

Option 4: Turn Off Interactive Mode in the NVIDIA Driver

Newer NVIDIA Linux Display Drivers offer an X configuration option that will disable various watchdog timer features built into the driver. The general format of the option is as follows:

Option "Interactive" "boolean"

It can be placed in the relevant "Device" section of the xorg.conf file for the X display. In order to disable, set the "boolean" parameter to "off". This option is documented in the release notes for the NVIDIA display driver (in drivers that support this feature) in the appendix describing the X config options. This option should rectify X hangs related to this issue, but CUDA performance will be less than optimal for the GPU since it is also handling X display tasks, and the X display updates will still be interrupted during CUDA kernel processing.

Note: Option 4 is only recommended for single GPU systems where both X and CUDA must run simultaneously and CUDA kernel execution cannot be bounded.

LIVE CHAT

Chat online with one of our support agents

ASK US A QUESTION

Contact Support for assistance

CHAT NOW

CONTACT US

Solutions: [Graphics Cards](#) | [GRID](#) | [High Performance Computing](#) | [Visualization](#) | [CUDA](#) | [Cool Stuff](#)

Corporate: [Events](#) | [Affiliate Program](#) | [Developers](#) | [Channel Partners](#) | [Careers](#) | [RSS Feeds](#) | [Newsletter](#) | [Contact Us](#) | [Security](#)

Copyright © 2017 NVIDIA Corporation [Legal Info](#) | [Privacy Policy](#)
