**Last Updated: Tuesday, 28 April 2020**

**Period search application with CUDA support**

**RELEASE NOTES**

**v102.15.0.1 - Tuesday, 28 April 2020**

Toggle Section

What's improved?

These improvements were released in v102.15.0.1:

* **Progress information**: Progress refresh status messaging now sends information to the client 10 times more often. This will remove the feeling that the application has stopped working or has been freeze..

What's fixed?

These issues were fixed in v102.15.0.1:

* **Slow performance on Windows 10 platforms**: Fixed issue with slow performance on Windows 10 platforms with respect to the Windows 7 and older “cuda55” applications.

**v102.14.0.1 - Saturday, 25 April 2020**

Toggle Section

What's improved?

These improvements were released in v102.14.0.1:

What's fixed?

These issues were fixed in v102.14.0.1:

* **100% CPU core (thread) utilization**: A bug was fixed where application causes 100% utilisation of whole CPU core (thread in hyperthreaded CPUs)

**v102.13.0.2 - Monday, 23 March 2020**

Toggle Section

What's improved?

These improvements were released in v101.13.0.2:

* **Added application info**: Added Application name and Version info to **stderr.txt** file.

What's fixed?

These issues were fixed in v102.13.0.2:

* **Hanging on devices with CC 7.5**: Fixed issue with hanging of the whole application on random tasks. There was a misaligned synchronisation of code inside the functions *mrqmin.cu* and *gauss\_errc.cu* that has been executed at threadIdx.x = 0 only which was leading to infinite loop of the calling kernel *CUDACalculateIter1\_mrqmin1\_end().*

**v102.12.0.1**

Toggle Section

What's improved?

These improvements were released in v102.12.0.1:

* **Added support for latest CC**: Application was built with latest CUDA SDK v100.2 adding support for latest NVIDIA devices with Compute Capabilities (CC). This build supports CC 3.0, 3.5, 3.7, 5.0, 5.2, 6.0, 6.1, 7.0, & 7.5 from Kepler, Maxwell, Pascal, Volta & Turing(1) architectures and their GeForce, Quadro, NVS and Tesla series, but does **NOT** supports Tegra and Jetson SoC Series devices (CC 3.2, 5.3, 6.2 & 7.2)(2,3) which are not subject of interest for this project.
* **Max User Registers**: The value of **Max Used Registers** is set to **32**. This way we achieved 100% kernel occupancy for the most time expensive kernels.
* **Message for unsupported CC**: New error message is introduced which now says supported CC limits of the application instead of old “CC2.0 and better supported only”.
* **Added CUDA version**: Added CUDA version info to **stderr.txt** file for the application.
* **Added GPU memory info**: Added GPU memory size and shared memory size info to **stderr.txt** file for current GPU card.

What's fixed?

These issues were fixed in v102.12.0.1:

* **Number of Blocks-per-SM**: Changed the number of Blocks-per-SM for CC 5.x according to the latest CUDA programing documentation.

Notes:

1. [CUDA (Compute Unified Device Architecture) – Wikipedia.](https://en.wikipedia.org/wiki/CUDA#GPUs_supported)
2. [CUDA applications originally developed for dGPUs attached to x86 systems may require modifications to perform efficiently on Tegra® systems.](https://docs.nvidia.com/cuda/cuda-for-tegra-appnote/index.html)
3. [Tegra is a system on a chip (SoC) series developed by Nvidia for mobile devices such as smartphones, personal digital assistants, and mobile Internet devices.](https://en.wikipedia.org/wiki/Tegra)