Abstract

GPUs and virtual machines have recently becoming popular computing platforms. While GPUs provide enormous computing power, virtual machines provide flexible resource management and utilization for organizations. Despite their advantages, there are only a handful research works that provide accesses to GPU for application programs running on virtual machines. In this research, we present the design and implementation of VirtualCUDA, a library and runtime system that allows accesses to GPU from virtual machines for CUDA applications. The main objectives of are to design and implement 1) a user-level library to pass CUDA command from virtual machines to GPU, and 2) the backend system to handle GPU resources. We have conducted a number of experiments to test our prototypes with the CUDA SDK matrix multiplication program. The first set of experiment evaluates the speed up of the application programs running VirtualCUDA against the serial programs running on virtual machines. We found that the Virtual CUDA program made substantial improvement over the serial one. In the next experiment, we demonstrate that the backend can handle multiple tasks at once; therefore increase resource utilization of the GPU. We have analyzed the experimental results and believe that VirtualCUDA has true practical values.