The para-virtualization technology has been pervasively deployed in many cloud computing platforms (e.g., Amazon EC2). This revolutionary trend also urges us to significantly reshaped modern operating systems to keep up the pace. Unfortunately, the current designs and implementations of modern operating systems lag behind the requirements.

In this paper, we focus on the improvement of I/O performance. ~~In specifically~~(Specifically), we aim to adopt OS kernel to further improve the I/O performance of all peripheral devices without scarifying the security of the paravirtualized platforms. By deeply analyzing modern Xen hypervisor and Linux kernel, we surprisingly notice that the page table updates (of guest OS) could cause IOMMU to flush IOTLB (in a granularity of page, which is for the sake of Xen’s security). However, these ~~unnecessary~~ flushes directly increase the miss rate of IOTLB, and consequently reduce I/O performance, especially for the high-speed devices. Note that we are the first one to ~~discover this dependence relationship~~ (explicitly claim that page table updates badly affect I/O performance). Based on this observation, ~~we propose a novel page table update algorithm to reduce the miss rate of IOTLB~~, (a novel algorithm is proposed to manage page table updates in guest kernel in order to reduce the miss rate of IOTLB while conform to Xen’s security policy). ~~and thereby improve overall I/O performance.~~

We implement our algorithm ~~on~~ (at the minimal customization of) modern Linux version 3.2.0 by (only) adding xxx SLoC, and evaluate the I/O performance in micro and macro ways. The micro experiment results indicate that the new algorithm is able to improve the I/O performance by effectively reducing the miss rate of IOTLB, especially when the page tables are frequently updated. The macro benchmarks shows that the I/O devices always produce better (or the same) performance, especially when the system frequently generate many temporal processes. (Also, the algorithm applies to other Linux versions above 3.2.0.)