論 文 の 内 容 の 要 旨

High Performance Solid-State Storage Systems with Memory-Aware Software Design
（メモリと制御ソフトウェアの統合設計による ソリッド・ステート・ストレージシステムの高性能化）

孫　超　　Chao　Sun

The market of the solid-state drives (SSDs) is expanding due to SSDs' higher performance and endurance while lower power consumption, compared with the hard disk drives (HDDs). Another driving force making SSDs as an alternative for HDDs is the continuously decreasing cost, thanks to the scaling and multi-bit technologies of the NAND flash memory. Compared to HDDs, the read performance of the SSDs is much faster due to the absence of seek time for the data access. Additionally, SSDs also have a significant sequential write performance advantage over HDDs. However, due to the "erase-before-write" characteristics and endurance problems of the NAND flash memory, the SSDs have little or even no advantage in random write performance than the HDDs.

In this dissertation, research is carried out to improve the SSD performance and endurance, reduce its energy consumption and cost. The storage class memory (SCM) and NAND flash capacity requirements are analyzed for the representative workloads with a proposed three-dimensional through-silicon-via (TSV) SCM/NAND flash hybrid architecture. The cost-effective configurations of combining the SCM and NAND flash is given according to the workload characteristics. On the other hand, a large SCM capacity is required for the hybrid SSD when the SCM is slow. In other words, there is a trade-off between the required SCM capacity and its access speed. Given the optimistic

and pessimistic SCM area cost models with a back-of-envelope estimation method, the SCM chip design with the lowest cost are obtained. Moreover, the design guidelines for the NAND flash block and page sizes are also provided.

A replacement algorithm cold data eviction (CDE) is proposed to use ReRAM as a write buffer for the NAND flash memory. In the hybrid SSD, the NAND like interface (I/F) is proposed for ReRAM according to a 50 ns HfO2 ReRAM measurement results. The data placement strategy is storing frequently accessed (hot) data and random data in ReRAM to exploit the spatial and temporal localities of the workload. When the ReRAM is almost full, the CDE algorithm aggregates and evicts the cold and less fragmented data to the NAND flash memory. Then the ReRAM space can be reclaimed for the new writes. Combined with the page-level mapping flash translation layer (FTL), 12.6-times performance improvement, 93% energy consumption reduction and 10-times endurance enhancement are achieved for a SSD workload from a financial server.

Furthermore, a NAND flash aware middleware called LBA scrambler is proposed to minimize the garbage collection (GC) overhead of the SSD. It is known that the GC is the bottleneck of the SSD write performance due to many page-copy operations for reclaiming the free space. The concept of the LBA scrambler is to intentionally write data to the fragmented pages in the next erase block. As a result, the page-copy overhead can be minimized. To achieve this, the SSD controller has to send the overwrite page hints to the LBA scrambler. With these hints, the LBA scrambler maps the logical block address (LBA) from the requests to the scrambler LBA (SLBA) and send the requests with SLBA to the SSD controller. With the LBA scrambler, maximum 4-times performance is enhanced.

Finally, a storage engine assisted SSD (SEA-SSD) is proposed for enhancing the SSD performance for the database applications. It is based on the idea that the layer in the host contains more rich information of the data activity. Since the SE manages the data written to the SSD, three kinds of hints are passed from the SE to the SSD controller. The SSD is divided into two segments for storing the dynamic and static data separately. Before the data's flushing to the storage, it is pre-judged as dynamic or static by the access pattern. According to this hint information, the data is written to the dynamic segment or static segment. Further, the data is also predicted to be the

dynamic data when it is firstly added to the flush list of the buffer pool. When the GC starts, this data is moved to the dynamic segment with hints. With the proposed SEA-SSD, over 20% performance improvement is achieved.