EED: Energy Efficient Disk Drive Architecture

Yuhui Deng       Frank Wang       Na Helian

Center for Grid Computing,
Cambridge-Cranfield High Performance
Computing Facilities, Cranfield University
Campus, Bedfordshire MK43 0AL,
United Kingdom

Department of Computer Science,
University of Hertfordshire,
United Kingdom

Information Sciences  2008
Outlines

• Introduction
• Architecture
• Implementation
• Experimental Evaluation
• Conclusions
Introduction (1/2)

• Energy saving has become one of the most important challenges in designing computing systems

• A number of non-volatile storage technologies are emerging and bring opportunities to the architecture design of disk drives
Motivation

– The emerging non-volatile storage technologies
– Highly skewed data access pattern

An Energy Efficient Disk (EED) drive architecture which can reduce energy consumption significantly, while also improving performance
Architecture (1/3)

• The storage space of the EED is divided into two areas in terms of the LBA
  – Magnetic disk media
  – Non-volatile memory

• Both areas are exclusive
Architecture (2/3)

• Categories of data
  – Hot data (Frequently accessed data)
  – Cold data (Rarely accessed data)

• Because most of the frequently accessed data can be satisfied from the non-volatile memory, the disk can be spun down and remain in the low power state
(a) Data layout of the EED before data migration

(b) Data layout of the EED after data migration
Implementation (1/6)

- Non-volatile memory
- Frequency monitoring
- Data migration
Implementation (2/6)

• Non-volatile memory
  – Integrating NAND

• Frequency monitoring
  – The correlations of data can be used to improve the effectiveness of storage caching
    • For example
      – Spatial locality
  – To keep the block correlations, clustering the consecutive data blocks into objects
Implementation (3/6) – Frequency monitoring

• An object is defined as a group of consecutive blocks in terms of LBA
  – Object reorganization is faster than block reorganization
  – It is easier to track the frequency of object use
Implementation (4/6) – Frequency monitoring

• Cache replacement algorithms
  – **Random Replacement (RR)** replaces cache lines by randomly selecting a cache line to evict
  – **Least Frequently Used (LFU)** evicts the cache lines least frequently
  – **Least Recently Used (LRU)** evicts the cache lines used least in the recent past

• Choosing **LRU**

• Using two fixed length LRU lists
  – A hot list
  – A recent list
Implementation (5/6) - Data migration

• Goal
  – Reorganizing data with minimal impact on the foreground workload

• A fixed reorganization interval presents a trade-off
  – If the interval is too short, frequent data reorganization may introduce too much overhead
  – If the interval is too long, performance degradation will be incurred as the data layout becomes less well adapted to the current data access patterns
Implementation (6/6) - Data migration

• For the EED, when the disk is idle, a process running in the background examines the hot list
  – If the objects on the hot list are stored in magnetic disk media, the objects will be moved to the non-volatile memory
  – When the non-volatile memory runs out of its 90% capacity, the objects which are not on the hot list will be migrated to the magnetic disk media
Experimental Evaluation (1/5)

• Three traces
  – Cello99 trace contains modern workloads which were collected in 1999
  – Cello96 trace was collected in 1996
  – TPC-D is an Oracle trace of decision support processes collected in 1997

• Disk specification
  – Quantum Atlas 10k disk
  – Samsung NAND flash memory (K9F6408U0A)
    • 8M x 8 Bit NAND Flash Memory
Experimental Evaluation (2/5)

• Data access pattern
• The optimal object size
• Evaluating energy saving
• Evaluating performance
Experimental Evaluation (3/5)

• Data access pattern

(a) Original data access pattern

(b) Access pattern after data reorganization
Experimental Evaluation (4/5)

- The optimal object size
  - Performance impact of differing object size
The graphs show the saved energy and performance variation as a function of the timeout threshold (seconds) for different datasets.

**Saved energy graph:**
- Cello99-EED (blue diamonds)
- Cello96-EED (pink squares)
- TPC-D-EED (black triangles)
- Cello99-original disk (green crosses)
- Cello96-original disk (purple asterisks)
- TPC-D-original disk (red circles)

**Performance variation graph:**
- Cello99 (blue diamonds)
- Cello96 (pink squares)
- TPC-D (green triangles)

The graphs illustrate how energy savings and performance variation change with increasing timeout thresholds for the datasets mentioned.
Conclusions

• The EED integrates a relatively small non-volatile flash memory into a traditional disk drive

• The EED reduces the number of program/erase calls and extending the life span of the flash memory