Impact of Flash Memory on Video-on-Demand Storage

Moonkyung Ryu
• Motivation
  – What is the problem?

• Background
  – Interval Caching
  – Flash Memory

• How to utilize Flash SSD?

• What kinds of Flash SSD is appropriate?

• What is the best strategy?

• Conclusion
Internet Video Rocks!

1995: Web overtakes Gopher, FTP
2000: P2P overtakes WWW
2013: Video content overtakes P2P
2025: Video communication overtakes content

Courtesy of CISCO
Internet Video Rocks!

YouTube and Hulu Traffic

PB/Month

Entire Internet Backbone in 2000 (US)  YouTube and Hulu Sites in 2008 (US)

Hulu

YouTube

Courtesy of CISCO
VoD Storage Requirements

- Video is continuous media data
- Real-time retrieval
- Large Bandwidth & Large Capacity

<table>
<thead>
<tr>
<th>Device</th>
<th>Random Read Throughput (I/O Request Size 1MB)</th>
<th>Cost per GB</th>
<th>Cost per MB/s</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM</td>
<td>&gt; 20GB/s</td>
<td>$23.0</td>
<td>$0.001</td>
</tr>
<tr>
<td>MLC SSD</td>
<td>&gt; 155MB/s</td>
<td>$1.88</td>
<td>$0.774</td>
</tr>
<tr>
<td>HDD15K</td>
<td>70MB/s</td>
<td>$1.23</td>
<td>$2.571</td>
</tr>
</tbody>
</table>

[Graph showing Aggr. Read Throughput (MB/s) vs. Number of concurrent sequential streams for MLC SSD and HDD15K]
VoD Storage Strategies

- HDD only
- HDD + DRAM
- HDD + Flash SSD
- Questions
  - How to utilize Flash SSD?
  - What kinds of Flash SSD is appropriate?
  - What is the best strategy?
Contents

• Motivation
  – What is the problem?

• Background
  – Interval Caching
  – Flash Memory

• How to utilize Flash SSD?

• What kinds of Flash SSD is appropriate?

• What is the best strategy?

• Conclusion
HDD + DRAM

- **File server or Web server**
  - Small file size
  - No real-time support
  - Block-level or File-level LRU (Least Recently Used)

- **Video server**
  - Large file size
  - Real-time support
    - via resource reservation
  - Interval Caching
    - Sequential video access
    - Caching the interval between two streams accessing the same video
Interval Caching

Video block sequences
Interval Caching

Video block sequences

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

S2

S1

B12

1 2 3

4 5 6

S1

S2
Interval Caching

Video block sequences

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Interval Caching

Video block sequences

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16
Interval Caching

Ordered buffer requirement list: B12 B22 B13 B14
Buffered streams: S12, S22, S13
Interval Caching

- **Beauty**
  - 100% cache hit for buffered streams
    - RAM space is reserved
    - Real-time support

- **Pitfalls**
  - No support for fast-rewind or fast-forward
    - Assume the same data rate for streams accessing the same video
Flash Memory

![Flash Memory Diagram]

3 States: Valid/Invalid/Clean

- Read
- Write

<table>
<thead>
<tr>
<th>Flash Type</th>
<th>Data Unit Size</th>
<th>Access Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Page (Bytes)</td>
<td>Block (Bytes)</td>
</tr>
<tr>
<td></td>
<td>Data</td>
<td>OOB</td>
</tr>
<tr>
<td>Small Block</td>
<td>512</td>
<td>16</td>
</tr>
<tr>
<td>Large Block</td>
<td>2048</td>
<td>64</td>
</tr>
</tbody>
</table>
Flash Memory

3 States: Valid/Invalid/Clean

<table>
<thead>
<tr>
<th>Flash Type</th>
<th>Data Unit Size</th>
<th>Access Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Page (Bytes)</td>
<td>Block (Bytes)</td>
</tr>
<tr>
<td></td>
<td>Data</td>
<td>OOB</td>
</tr>
<tr>
<td>Small Block</td>
<td>512</td>
<td>16</td>
</tr>
<tr>
<td>Large Block</td>
<td>2048</td>
<td>64</td>
</tr>
</tbody>
</table>
# Flash Memory

3 States: Valid/Invalid/Clean

<table>
<thead>
<tr>
<th>Flash Type</th>
<th>Data Unit Size</th>
<th>Access Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Page (Bytes)</td>
<td>Block (Bytes)</td>
</tr>
<tr>
<td></td>
<td>Data</td>
<td>OOB</td>
</tr>
<tr>
<td>Small Block</td>
<td>512</td>
<td>16</td>
</tr>
<tr>
<td>Large Block</td>
<td>2048</td>
<td>64</td>
</tr>
</tbody>
</table>

The diagram shows the components of an SSD (Solid State Drive), including the File System, RAM Buffer, Write Buffer, Controller (FTL), NAND Flash Memories, and the process of logical sector read and write, along with valid, invalid, and clean states.
## Flash Memory

3 States: Valid/Invalid/Clean

<table>
<thead>
<tr>
<th>Flash Type</th>
<th>Data Unit Size</th>
<th>Access Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Page (Bytes)</td>
<td>Block (Bytes)</td>
</tr>
<tr>
<td></td>
<td>Data</td>
<td>OOB</td>
</tr>
<tr>
<td>Small Block</td>
<td>512</td>
<td>16</td>
</tr>
<tr>
<td>Large Block</td>
<td>2048</td>
<td>64</td>
</tr>
</tbody>
</table>
Flash Memory

3 States: Valid/Invalid/Clean

<table>
<thead>
<tr>
<th>Flash Type</th>
<th>Data Unit Size</th>
<th>Access Time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Page (Bytes)</td>
<td>Block (Bytes)</td>
</tr>
<tr>
<td></td>
<td>Data</td>
<td>OOB</td>
</tr>
<tr>
<td>Small Block</td>
<td>512</td>
<td>16</td>
</tr>
<tr>
<td>Large Block</td>
<td>2048</td>
<td>64</td>
</tr>
</tbody>
</table>
Flash Memory

- **Constraints**
  - Read/Write in a PAGE unit
  - Erase in a BLOCK unit
  - No in-place update
    - No write on a page unless it’s clean
  - Limited erase count
    - SLC (100,000), MLC (10,000)
    - Wear-leveling

- **Flash Translation Layer**
  - Mapping table on SRAM
    - Virtual address (exposed to upper level)
    - physical address (on flash chips)
  - Garbage Collection
    - Log structured mechanism
    - Hide erase operation to upper level
• Motivation
  – What is the problem?
• Background
  – Interval Caching
  – Flash Memory
• How to utilize Flash SSD?
• What kinds of Flash SSD is appropriate?
• What is the best strategy?
• Conclusion
Flash Memory

- Fast Random Read and Sequential Write

![Graph showing throughput vs. request size for SAMSUNG MMDOE56G5MXP-0VB MLC 256GB Flash Memory with different read and write patterns.](image-url)
• Cheap MLC Flash Memory

<table>
<thead>
<tr>
<th>Device</th>
<th>Seq. Read (MB/s)</th>
<th>Rand. Read (MB/s)</th>
<th>$ / GB</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM</td>
<td>&gt; 20480</td>
<td>&gt; 20480</td>
<td>23.00</td>
</tr>
<tr>
<td>HDD15K</td>
<td>160</td>
<td>70</td>
<td>1.23</td>
</tr>
<tr>
<td>MLC SSD</td>
<td>&gt; 155</td>
<td>&gt; 155</td>
<td>1.88</td>
</tr>
</tbody>
</table>
• Highly skewed video access pattern

Flash Memory

• Reasons to go for Flash Memory
  – Fast Random Read
  – Fast Sequential Write
  – Cheap MLC Flash Memory
  – Skewed video access pattern
  – Very Low Power

• Problems
  – Higher cost/GB than HDDs
  – Poor small random write
  – Unpredictable garbage collection
  – Limited life time
How to utilize Flash Memory?

- File-level LFU
  - Guarantee cache hit
  - Easy to guarantee continuous delivery

- Flash Memory for B/W and Capacity
- HDD for Capacity
- Power reduction
Contents

• Motivation
  – What is the problem?
• Background
  – Interval Caching
  – Flash Memory
• How to utilize Flash SSD?
• What kinds of Flash SSD is appropriate?
• What is the best strategy?
• Conclusion
**Which Flash Memory SSD?**

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>MODEL</td>
<td>RiData NSSD-S25-64-C06MPN</td>
<td>SAMSUNG MMDOE56G5MXP-0VB</td>
<td>INTEL X25-M G1</td>
<td>Fusion-io ioDRIVE</td>
</tr>
<tr>
<td>CLASS</td>
<td>LOW</td>
<td>LOW</td>
<td>HIGH</td>
<td>ENTERPRISE</td>
</tr>
<tr>
<td>TYPE</td>
<td>MLC</td>
<td>MLC</td>
<td>MLC</td>
<td>SLC</td>
</tr>
<tr>
<td>CAPACITY (GB)</td>
<td>64</td>
<td>256</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>SEQ. READ B/W (MB/s)</td>
<td>155</td>
<td>220</td>
<td>250</td>
<td>750</td>
</tr>
<tr>
<td>$/GB</td>
<td>1.88</td>
<td>2.34</td>
<td>2.50</td>
<td>61.95</td>
</tr>
</tbody>
</table>

Random Write throughput when request size is 4KB.
Which Flash Memory SSD?

Is High-end SSD good for the architecture?  NO!

Random Read and Sequential Write operations are mixed by 1:1. Request size is 1MB.

Average Random Read throughput without write operations

Average and Standard deviation of Random Read throughput with write operations
Contents

• Motivation
  – What is the problem?
• Background
  – Interval Caching
  – Flash Memory
• How to utilize Flash SSD?
• What kinds of Flash SSD is appropriate?
• What is the best strategy?
• Conclusion
File-level LFU v.s. Interval Caching

- **Assumptions**
  - 300 videos in total
  - Homogeneous constant bitrate 245KB/s
  - Homogeneous video length
  - Popular videos are cached in advance

- **Control Parameters**
  - Arrival rate (req/sec)
  - Video popularity (Zipf distribution)
  - Video length

- **Performance metrics**
  - Number of buffered streams
  - Hit ratio
  - Rejection probability

<table>
<thead>
<tr>
<th>Memory Type</th>
<th>Capacity</th>
<th>Transfer Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLASH MEMORY</td>
<td>64GB</td>
<td>155MB/s</td>
</tr>
<tr>
<td>RAM</td>
<td>5.23GB</td>
<td>Infinite</td>
</tr>
<tr>
<td>File-level LFU</td>
<td>584GB</td>
<td>100MB/s</td>
</tr>
<tr>
<td>Interval Caching</td>
<td>584GB</td>
<td>100MB/s</td>
</tr>
</tbody>
</table>
Zipf 0.271, Video Length 60mins

File-level LFU v.s. Interval Caching

![Graphs showing number of buffered streams and rejection probability vs. arrival rate for RAM and FLASH.](image)
File-level LFU v.s. Interval Caching

Zipf 0.271, Arrival Rate 0.16
What is the best strategy?

- Strategies
  - HDD only
  - HDD + DRAM (Interval Caching)
  - HDD + MLC Flash SSD (File-level LFU)

- Total cost of ownership =
  total cost of acquisition +
  total cost of operation
What is the best strategy?

- Simulation Parameters
  - Video Library: 300
  - Zipf (Video Popularity): 0.271
  - Video Bitrate: 245 KB/s
  - Video Length: 60 min
  - Arrival Rate: 1 req/sec
  - Rejection Probability: 0%

- Device profile

<table>
<thead>
<tr>
<th>Device</th>
<th>Capacity (GB)</th>
<th>Rand. Read (MB/s)</th>
<th>$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DRAM</td>
<td>1</td>
<td>∞</td>
<td>23</td>
</tr>
<tr>
<td>HDD15K</td>
<td>146</td>
<td>70</td>
<td>180</td>
</tr>
<tr>
<td>MLC SSD</td>
<td>64</td>
<td>155</td>
<td>120</td>
</tr>
</tbody>
</table>

3600 concurrent streams
What is the best strategy?

- **Minimum cost configuration**

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Configuration</th>
<th>Streams / $</th>
</tr>
</thead>
<tbody>
<tr>
<td>HDD only</td>
<td>13</td>
<td>1.46</td>
</tr>
<tr>
<td>HDD + DRAM (Interval Caching)</td>
<td>2 + 12</td>
<td>1.54</td>
</tr>
<tr>
<td>HDD + Flash SSD (File-level LFU)</td>
<td>5 + 2</td>
<td>3.55</td>
</tr>
</tbody>
</table>

![Bar chart showing stream / $ for different configurations](chart.png)
• NAND Flash Memory is promising

• Questions
  – How to utilize Flash SSD?
    • Avoid random write operations
    • File-level LFU
  – What kinds of Flash SSD is appropriate?
    • Low-end MLC Flash SSD is better
  – What is the best strategy?
    • HDD+MLC Flash SSD
Thank You!