Join Processing for Flash SSDs: Remembering Past Lessons

Jaeyoung Do, Jignesh M. Patel

Department of Computer Sciences
University of Wisconsin-Madison
Flash Solid State Drives (SSDs)

- **Benefits of Flash SSDs**
  - No seek time, small latency

- **Flash Densities**
  - 1995. 16 MB NAND Flash
  - 2005. 16 GB NAND Flash
  - 2010. 1 TB NAND Flash (predicted by Samsung)

- **Flash Prices**
  - Decreasing continuously
Flash SSDs for DBMSs

Many previous works with flash SSDs

- Flash-based DBMS [Gray ACM Queue 08, Graefe DaMoN07]
- In-Page Logging [Lee VLDB07]
- Transaction Processing [Lee VLDB08]
- B+ Tree Index [Li ICDE09]
- I/O Benchmarks [Bouganim CIDR09]

We focus on **Join algorithms**

- New join algorithms [Shah DaMoN08, Tsirogiannis SIGMOD09]

What lessons learnt for magnetic HDDs still apply to flash SSDs?
Goals

► Demonstrate the importance of recalling past lessons about efficient joins in magnetic HDDs

► Explore the effects of various parameters for joins on flash SSDs

1. Not inventing new join algorithms

2. Providing better insights for join performance on flash SSDs
Our Approach

▶ Investigate four popular ad hoc join algorithms
  • Block Nested Loops Join (BNL)
  • Sort-Merge Join (SM)
  • Grace Hash Join (GH)
  • Hybrid Hash Join (HH)

▶ Conduct experiments as varying various parameters
  • Memory buffer pool size
  • Page size
  • I/O unit size
Assumptions

- Blocked I/O is available

- We use the buffer allocation strategy tuned for magnetic HDDs [Haas et al. VLDB97]
Block Nested Loops Join

$I_S = \left[ \sqrt{y|S| (y|S| + B(y + |S|)) - y|S|} \right] / (y + |S|), \quad I_R = B - I_S$
Sort-Merge Join

Buffer Pool
- Working Space
- Input Buffer
- Output Buffer

Disk
- R
- S

Disk

Buffer Pool
- Input Buffer
- Input Buffer
- Input Buffer
- Result
Grace Hash Join

Buffer Pool

Input Buffer
Bucket 1
... 
Bucket k

Disk

Input Buffer for R

Input Buffer for S

Disk

Result
Hybrid Hash Join

Buffer Pool

In Memory Bucket

Input Buffer

Bucket 1

Bucket k

Disk

Result

Buffer Pool

Input Buffer for R

Input Buffer for S

Disk

Result

DaMoN 09
Roadmap

- Introduction / Goals
- Ad-hoc join algorithms
- Experimental Results
- Conclusion / Future Work
Experimental Setup

- A single-thread and light-weight database engine
- Flash SSD and magnetic HDD
  - OCZ Core Series 2.5” SATA 60 GB
  - TOSHIBA 5400 RPM 320 GB
- Data Set: TPC-H
  - Customer: 730 MB
  - Orders: 5 GB
- Platform
  - Dual Core 3.2 GHz Intel Pentium, Red Hat
  - Max. DB buffer pool size: 600 MB

<table>
<thead>
<tr>
<th>SSD</th>
<th>HDD</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Avg. Seek Time</td>
</tr>
<tr>
<td>0.35 ms</td>
<td>Avg. Latency</td>
</tr>
<tr>
<td>120 MB/sec</td>
<td>Read Data Transfer Rate</td>
</tr>
<tr>
<td>80 MB/sec</td>
<td>Write Data Transfer Rate</td>
</tr>
<tr>
<td>230.99 $ (3.85 $/GB)</td>
<td>Price</td>
</tr>
</tbody>
</table>

Source: OCZ and TOSHIBA
Effect of Blocked I/O (HDD)

500 MB Buffer Pool, 8 KB Page Size

Non-Blocked I/O

Blocked I/O

Join Time (sec)

CPU time

I/O time

2.1X

2.2X

2.0X

2.3X

BNL  SM  GH  HH

BNL  SM  GH  HH

DaMoN 09
Effect of Blocked I/O (SSD)

500 MB Buffer Pool, 8 KB Page Size

Non-Blocked I/O       Blocked I/O

Join Time (sec)

<table>
<thead>
<tr>
<th>SSD</th>
<th>BNL</th>
<th>SM</th>
<th>GH</th>
<th>HH</th>
<th>BNL</th>
<th>SM</th>
<th>GH</th>
<th>HH</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Using blocked I/O is critical

CPU time
I/O time

1.7X
1.6X
1.9X

DaMoN 09
Joins are I/O Bound? (HDD)

8 KB Page Size, Blocked I/O

Join Time (sec)

200 MB

Buffer Pool Size

500 MB

HDD

BNL SM GH HH

Join Time (sec)

0

0.31

0.58

0.69

0.61

0.34

0.65

0.62

0.68

CPU time

I/O time

DaMoN 09
Joins are I/O Bound? (SSD)

8 KB Page Size, Blocked I/O

Join Time (sec)

SSD

BNL  SM  GH  HH

200 MB  500 MB

Buffer Pool Size

CPU time

I/O time

Joins may become CPU-bound sooner
Effect of Varying the Page Size (SSD)

More details in our DaMoN’09 paper

When using Blocked I/O, the page size has a small impact on join performance
Conclusions / Future Work

▶ Traditional Join optimizations continue to be important with flash SSDs
  • Blocked I/O dramatically improves join performance
  • Buffer allocation strategy has an impact on join performance
  • It is even more critical to consider both CPU and I/O costs

▶ Future Work
  • Expand the range of hardware, and consider other HDD-based configurations
  • Derive detailed cost models for existing join algorithms, and explore the optimal buffer allocations for flash SSDs