236834 - Seminar on Storage Systems: Deduplication

File 1

File 2

File 3

Spring 2017

Computer Science Department
Administration

• Mandatory attendance
• Online questionnaire before every lecture
• Paper presentation
  – Understanding of the paper
  – Slides quality
  – Presentation
• Participation in others’ lectures
  ➔ If you haven’t already, pair up and select papers today
Reminder: Operating Systems

• The operating system exports storage devices as “drives”
• Drive \( i \) appears as an array of \( n_i \) blocks
• The OS is responsible for mapping files and offsets to blocks
• Optional per-device mapping by controller
• Devices store blocks (pages) of data (block devices)

• Data has to be in main memory (RAM) to be used for computation

• The main memory is organized into page-sized buffers

• The memory allocated for data blocks is much smaller than the devices’ capacity

• Blocks are evicted and their content should be written back to the device
Hard Disk Drive (HDD)
Sector read time = seek time + rotational delay + transfer time
Disk Addresses

• Physical organization:
  plate, track, sector

→ Constant arm movements

• Logical organization:
  cylinder, track, sector

→ Less arm movements for sequential access
A typical HDD (2012)

Size: 3 platters = 6 surfaces
184,000 tracks per surface
1600 sectors per track on average, 512 bytes per sector

Capacity:
- Sector: 512B
- Track: 1600 x 512B = 800KB
- Cylinder: 800 KB x 6 = 4800KB = 4.7 MB
- Enclosure: 4.7MB x 184,000 = 866250 MB = 846 GB

Performance:
- Rotational delay: 10,000 RPM = 167 rounds/s
  \[ T_{\text{rot}} = 6 \text{ msec} \]
- Seek time: 0.2 msec (minimum)
  7.7 msec (maximum)
  \[ 3.7 \text{ msec (average)} \]
- Transfer time: 3.75 \( \mu \text{sec} \) (average sector)
Writing and Updating

• The disk head writes full sectors
  – A first write takes the same time as a read
  – Overwriting an entire sector is equivalent
  – Partial overwrite is an update

• When the sector’s content is not in memory
  – Calculate sector address
  – Read the sector into a buffer
  – Update the sector in memory
  – What until the head reaches the sector again
  – Write (transfer)

→ Equivalent to read time + one rotation
What about SSDs?

- **Pages** (4KB-32KB) are the **read/write** unit [100 μsec scale]
- **Blocks** (at least 128 pages) are the **erase** unit [msec scale]
- Hundreds/thousands of blocks per plane
- Several planes per chip/die
- More details in a few weeks...
Reminder: File Systems

- Mapping layer between application and storage
- Manages space in one device

→ Allocate contiguous space to large files
→ Minimize overhead of metadata management
File System Access

• Read:
  – Read inode
  – Optionally read indirect mapping blocks
  – Read block

• Append:
  – Allocate another block
  – Optionally allocate another mapping block
Snapshots (Copy-on-Write)

- Different versions of a file share blocks
- Good for backup and transactions

Slide by Kalpak Shah, Lustre Group, Sun Microsystems Inc.
What About Different Files?

• May also share blocks
  – Attachments
  – Configuration files
  – Company logo and other headers

• On shared storage: same-same-but-different
  – Common system files, application binaries
  – Shared documents, code development
  – Popular photos, videos, etc.
  – Virtual machine images

→ Deduplication!
Backups on EMC’s DataDomain

trade-off

noun
noun: tradeoff

A balance achieved between two desirable but incompatible features; a compromise.
"A trade-off between objectivity and relevance"

Translations, word origin, and more definitions

- Risk vs. profit
- Cost vs. benefit
- Quantity vs. quality
- Time vs. energy

→ Inode size
→ Number of snapshots
“Naïve” Deduplication

For each new file

Compare each block to all existing blocks

If new, write block and add pointer
If duplicate, add pointer to existing copy

Are we done?
Identifying Duplicates

• It’s unreasonable to “Compare each block to all existing blocks”
  ➔ Fingerprints
    Cryptographic hash of block content
    Low collision probability

• It’s also unreasonable to compare to all fingerprints...
  ➔ Fingerprint cache

• What are the tradeoffs?
Reading and Restoring

• How long does it take to read File1?
• How long does it take to read File3?

• What are the tradeoffs?
Distributed Storage

Increase storage capacity and performance with multiple storage servers

• Each server is a separate machine (CPU,RAM,HDD/SSD)

• Data access is distributed between servers

❖ **Scalability**
   Increase capacity with data growth

❖ **load balancing**
   Independent of workload

❖ **failure handling**
   Network, nodes and devices always fail
Distributed Deduplication

- Where/when should we look for duplicates?
- Where should we store each file?

- What are the tradeoffs?
Roadmap

• Fundamental deduplication principles
• Backup vs. primary storage
• Local vs. distributed storage and deduplication
• Domain specific: virtual storage, SSDs
• Chunking and duplicate detection
Presentation Guidelines

• One paper per week (two hours)
• Plan for 10 minutes of discussion at the end

• General guidelines (for this seminar and for life)
  http://users.ece.cmu.edu/~pueschel/teaching/guides/guide-presentations.pdf

• Using existing slides
  – Not encouraged
  – If unavoidable, reference and give proper credit
Presentation Checklist

• What is the problem?
  – What is the motivation for deduplication in this system?
  – What is the optimization objective?
  – Why are previous solutions not good enough?

• How does it work?
  – What happens during a read, write and delete?
  – Of duplicate and unique data
  – Hint: use examples here

• How is this similar to or different from previous systems presented in this seminar?
  – Domain, objectives, assumptions

• What are the tradeoffs?
  – Are they addressed in the paper?
  – Are they reasonable?
Chunking

- Chunking: splitting files into blocks
- Fixed-size chunks: usually aligned to device blocks
- What is the best chunk size?
Updates and Versions

- **Best case:**
  
  -aabbccdd
  
  → aAabbccdd

- **Worst case:**
  
  -aabbccdd
  
  → aAabbccddd

Ideally...
Variable-Size Chunks

• Basic idea: chunk boundary is a random string
• For example: 010
• \text{aa010bb010cc010dd} \rightarrow \text{aAa010bb010cc010dd}
• Boundaries should be:
  – Not too short/long
  – Not too popular (000000...)
  – Easy to identify
Fingerprints

- A set of functions $F = \{f: \Omega \rightarrow \{0,1\}^k\}$

- For a set $S \subset \Omega$, $|f(S)| = |S|$

- $f(A) \neq f(B) \Rightarrow A \neq B$

- $A \neq B \Rightarrow f(A) \neq f(B)$ with high probability
Rabin Fingerprints

• A binary string is a representation of a polynomial over $\mathbb{Z}_2$
  
  \[ A = (1,1,0,1,0,0,0,1) \rightarrow A(t) = t^7 + t^6 + t^4 + 1 \]

• $f(A) = A(t) \mod P(t)$, $P$ irreducible polynomial of degree $k$
  
  \[ P(t) = t^3 + t + 1 \text{ (here } k = 3) \]
  
  \[ f(A) = (t^7 + t^6 + t^4 + 1) \mod (t^3 + t + 1) = t + 1 \]

  \[ t^7 + t^6 + t^4 + 1 = (t^4 + t^3 + t^2 + t)(t^3 + t + 1) + t + 1 \]

  \[ f(A) = (0,1,1) \]

• Probability of collision ($A \neq B$ but $f(A) = f(B)$) is $2^{-k} = 1/8$
Computing Rabin Fingerprints

- \( f(A_{1,L}) = (a_1 t^{L-1} + a_2 t^{L-2} + \cdots + a_L) \mod P \)
- \( f(A_{2,L+1}) = (a_2 t^{L-1} + a_3 t^{L-2} + \cdots + a_{L+1}) \mod P = (t \cdot A_{1,L} + a_{L+1} - a_1 t^L) \mod P = (t \cdot f(A_{1,L}) + a_{L+1} - a_1 t^L) \mod P \)

\[ A_{1,8} = (1,1,0,1,0,0,0,1) \rightarrow A_{1,8}(t) = t^7 + t^6 + t^4 + 1 \]
\[ A_{2,9} = (1, (1,0,1,0,0,0,1,1)) \rightarrow A_{2,9}(t) = t^7 + t^5 + t + 1 = t \cdot A_{1,8}(t) + 1 - t^8 \]

- **Direct computation:** \( f(A_{2,9}) = (t^7 + t^5 + t + 1) \mod (t^3 + t + 1) = t^2 + 1 \)

- **Rolling hash:**
  \[
  f(A_{2,9}) = (t \cdot A_{1,8}(t) + 1 - t^8) \mod (t^3 + t + 1) \\
  = (t \cdot (t + 1) + 1 - t^8) \mod (t^3 + t + 1) \\
  = (t^2 + t + 1 - t^8) \mod (t^3 + t + 1) \\
  = (t^2 + t + 1 - t) \mod (t^3 + t + 1) = t^2 + 1 \\
  f(A_{2,9}) = (1,0,1) \\
  \boxed{(t^8) \mod (t^3 + t + 1) = t} \]
Precomputed Tables

• In practice we shift by bytes (2 bits in this example)
  \[ A_{1,8} = (1,1,0,1,0,0,0,1) \rightarrow A_{1,8}(t) = t^7 + t^6 + t^4 + 1 \]
  \[ A_{3,10} = (1,1,0,1,0,0,0,1,1,0) \rightarrow A_{3,10}(t) = t^6 + t^2 + t \]
  \[ = t^2 A_{1,8}(t) + t - (t^9 + t^8) \]

• The rolling hash still requires computations
  \[
  f(A_{3,10}) = (t^2 A_{1,8}(t) + t - (t^9 + t^8)) \mod (t^3 + t + 1) \\
  = (t^2(t + 1) + t - (t^9 + t^8)) \mod (t^3 + t + 1) \\
  = (t^3 + t^2 + t - (t^9 + t^8)) \mod (t^3 + t + 1) \\
  = (t^3 + t^2 + t - (t^2 + t)) \mod (t^3 + t + 1) = t^3 \mod (t^3 + t + 1) = t + 1
  \]

\[
  f(A_{3,10}) = (0,1,1)
  \]

<table>
<thead>
<tr>
<th>(a_1)</th>
<th>(a_2)</th>
<th>(a_1 t^9 + a_2 t^8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>(t)</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>(t^2)</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>(t^2 + t)</td>
</tr>
</tbody>
</table>
857 Desktops at Microsoft

Figure 1: Deduplication vs. Chunk Size for Various Algorithms

Figure 2: Deduplication vs. Deduplication Domain Size