RAID
Redundant Array of Inexpensive Disks

CMPU 334 – Operating Systems
Jason Waterman
RAID (Redundant Array of Inexpensive Disks)

• Use multiple disks in concert to build a faster, bigger, and more reliable disk system
  • RAID just looks like one big disk to the host system

• Advantages
  • Performance & Capacity: Using multiple disks in parallel
  • Reliability: RAID can tolerate the loss of a disk

RAIDs provide these advantages transparently to systems that use them
RAID Interface

• When a RAID receives I/O request:
  1. The RAID calculates which disk to access
  2. The RAID issue one or more physical I/Os to do so

• RAID example: A mirrored RAID system
  • Keep two copies of each block (each one on a separate disk)
  • Perform two physical I/Os for every one logical I/O it is issued
RAID Internals

• A microcontroller
  • Run firmware to direct the operation of the RAID

• Volatile memory (such as DRAM)
  • Buffer data blocks

• Non-volatile memory
  • Buffer writes safely

• Specialized logic to perform parity calculation

• In essence, a RAID is a specialized computer system!
Fault Model

• RAIDs are designed to detect and recover from certain kinds of disk faults

• Assume for now a fail-stop fault model
  • A disk can be in one of two states: Working or Failed
    • Working: all blocks can be read or written
    • Failed: the disk is permanently lost
  • RAID controller can immediately observe when a disk has failed

• Worry about other types of failures later
  • Disk corruption
  • Having a single block of the disk fail in a otherwise working drive
How to evaluate a RAID

• Capacity
  • How much useful capacity is available to systems?

• Reliability
  • How many disk faults can the given design tolerate?

• Performance
  • Throughput in MB/s
  • Challenging to evaluate as it heavily depends on workload
  • Talk more in depth later
RAID Designs

• RAID level 0 (striping)
  • No redundancy
  • Gives an upper bound on performance and capacity

• RAID level 1 (mirroring)
  • Tolerates disk failures
  • Keeps one or more copies of each block

• RAID level 4/5 (parity-based redundancy)
  • More complicated than mirroring
  • Provides redundancy with better space utilization than mirroring
RAID Level 0: Striping

- RAID Level 0 is the simplest form as striping blocks
  - Spread the blocks across the disks in a round-robin fashion
  - **No redundancy**
  - Excellent performance and capacity

<table>
<thead>
<tr>
<th>Disk 0</th>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
</tr>
</tbody>
</table>

**RAID-0: Simple Striping**
(Assume here a 4-disk array)
RAID Chunk Size

• Chunk size: number of blocks per disk in a stripe
  • Example: RAID-0 with a bigger chunk size and 4KB blocks
  • Chunk size: 2 blocks (8 KB)
  • A Stripe: 4 chunks (32 KB)

• Chunk size affects performance
  • Small chunk size
    • Many files get striped across many disks
    • Increases parallelism for a single file
    • Increases positioning time to access blocks
  • Large chunk size
    • Reduces intra-file parallelism
    • Reduces positioning time

Determining the “best” chunk size is hard to do
Most arrays use larger chunk sizes (e.g., 64 KB)
Evaluating RAID Performance

• Consider two performance metrics
  • Single request latency (time to process a single read request)
  • Steady-state throughput (total bandwidth of many concurrent requests)
    • Main focus of our analysis

• Workload
  • **Sequential**: e.g., access 1MB of data (block (B) to block (B + 1MB))
  • **Random**: e.g., access 4KB at random logical address

• A disk can transfer data at
  • \( S \) MB/s under a sequential workload
  • \( R \) MB/s under a random workload
Sequential vs Performance Calculation

• Sequential (S) vs random (R)
  • **Sequential**: transfer 10 MB on average as continuous data
  • **Random**: transfer 10 KB on average
  • Average seek time: 7 ms
  • Average rotational delay: 3 ms
  • Transfer rate of disk: 50 MB/s

• Results:
  • \[ S = \frac{\text{Amount of Data}}{\text{Time to access}} = \frac{10 \text{ MB}}{210 \text{ ms}} = 47.62 \text{ MB/s} \]
  • \[ R = \frac{\text{Amount of Data}}{\text{Time to access}} = \frac{10 \text{ KB}}{10.195 \text{ ms}} = 0.981 \text{ MB/s} \]
Evaluating RAID-0 Performance

• Single request latency
  • Similar to that of a single disk
  • RAID-0 simply redirects the request to the proper drive

• Steady-state throughput
  • Expect to get the full bandwidth of the system
  • **Sequential** workload: \( N \cdot S \) MB/s
    • \( S \) is the sequential bandwidth of a single disk
  • **Random** workload: \( N \cdot R \) MB/s
    • \( R \) is the random bandwidth of a single disk

• Represent the upper-bound on RAID performance
RAID Level 0 Analysis

• **Capacity** → RAID-0 is optimal
  • Striping delivers $N$ disks worth of useful capacity

• **Performance** of striping → RAID-0 is excellent
  • All disks are utilized often in parallel

• **Reliability** → RAID-0 is bad
  • Any disk failure will lead to data loss

$N$ : the number of disks
RAID Level 1: Mirroring

- RAID Level 1 tolerates disk failures
  - Copy more than one of each block in the system
  - Block copies are placed on a separate disk

- Two choices
  - RAID-10 (RAID 1+0): mirrored pairs and then stripe (more common, shown on the left)
  - RAID-01 (RAID 0+1): contain two large striping arrays, and then mirrors (shown on the right)
RAID-1 Analysis

• **Capacity**: RAID-1 is Expensive
  • The useful capacity of RAID-1 is $N/2$

• **Reliability**: RAID-1 does well
  • It can tolerate the failure of any one disk (up to $N/2$ failures depending on which disks fail)

$N$ : the number of disks
Performance of RAID-1

• Single request latency
  • Reads have the latency of a single disk
  • Writes need two physical writes to complete
    • It suffers the worst-case seek and rotational delay of the two requests

• Steady-state throughput
  • **Sequential Write**: $\frac{N}{2} \cdot S$ MB/s
    • Each logical write must result in two physical writes
  • **Sequential Read**: $\frac{N}{2} \cdot S$ MB/s
    • Each disk will only deliver half its peak bandwidth
    • This seems counterintuitive, why is it so?
  • **Random Write**: $\frac{N}{2} \cdot R$ MB/s
    • Each logical write must turn into two physical writes
  • **Random Read**: $N \cdot R$ MB/s
    • Distribute the reads across all the disks

<table>
<thead>
<tr>
<th></th>
<th>Disk 0</th>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

RAID-1: Sequential read request (0-7)
RAID Level 4: Saving Space With Parity

• Add a single **parity block**
  • A parity block stores the redundant information for that stripe of blocks

<table>
<thead>
<tr>
<th>Disk 0</th>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
<th>Disk 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>P0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>P1</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>P2</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>P3</td>
</tr>
</tbody>
</table>

* P: Parity

Five-disk RAID-4 system layout
RAID Level 4 (Cont.)

• **Compute parity**: the XOR of all of bits

<table>
<thead>
<tr>
<th>C0</th>
<th>C1</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>X</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

\[ \text{XOR}(0,0,1,1) = 0 \]
\[ \text{XOR}(0,1,0,0) = 1 \]

• Counting the parity bit, the row will always have an **even number of ones**

• **Recover from parity**
  
  • Imagine the bit of the C2 in the first row is lost
    1. Reading the other values in that row: 0, 0, 1
    2. The parity bit is 0 \(\rightarrow\) even number of ones in the row
    3. The missing data must be a 1

• **For a block of data**
  
  • Compute parity bits for \(b_0b_1b_2...b_{511}\)
RAID-4 Analysis

• **Capacity**
  • The useful capacity is \( (N - 1) \)

• **Reliability**
  • RAID-4 tolerates **1 disk failure** and no more

\( N \) : the number of disks
RAID-4 Analysis (Cont.)

• Performance
  • Steady-state throughput
    • Sequential read: \((N - 1) \cdot S\) MB/s
    • Sequential write: \((N - 1) \cdot S\) MB/s (using full-stripe write optimization)
      • Parity block can be written in parallel with the data blocks
  • Random read: \((N - 1) \cdot R\) MB/s
    • Reads are spread across the disks, but not the parity disk

<table>
<thead>
<tr>
<th>Disk 0</th>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
<th>Disk 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>P0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>P1</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>P2</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>14</td>
<td>15</td>
<td>P3</td>
</tr>
</tbody>
</table>

Full-stripe Writes In RAID-4
Random write performance for RAID-4

• Overwrite a block + update the parity

• **Method 1: additive parity**
  • Read in all of the other data blocks in the stripe (N reads)
  • XOR those blocks with the new block
  • Data and parity writes can happen in parallel (2 writes)
  • **Problem:** larger RAIDs require a high number of reads to compute parity
**Random write performance for RAID-4 (Cont.)**

- **Method 2**: *subtractive parity*

  
<table>
<thead>
<tr>
<th>C0</th>
<th>C1</th>
<th>C2</th>
<th>C3</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>XOR(0,0,1,1)=0</td>
</tr>
</tbody>
</table>

  - Update C2(old) → C2(new)
    1. Read in the old data at C2 (C2(old)=1) and the old parity (P(old)=0)
    2. Calculate P(new): \[ P(new) = (C2(old) \ \text{XOR} \ C2(new)) \ \text{XOR} \ P(old) \]
       - If C2(new)==C2(old) → P(new)==P(old)
       - If C2(new)!==C2(old) → Flip the old parity bit
  - Two reads (C2(old) and P(old)) and two writes (C2(new) and P(new))
    - Regardless of the size of the array
A I/O latency in RAID-4

• **A single read**
  • Equivalent to the latency of a single disk request

• **A single write** (using subtractive parity)
  • Two reads and then two writes
    • Data block + Parity block
    • The reads and writes can happen in parallel
  • Total latency is about twice that of a single disk
Small-write problem

- The parity disk can be a **bottleneck**
  - Example: update blocks 4 and 13 (marked with *)

<table>
<thead>
<tr>
<th>Disk 0</th>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
<th>Disk 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>P0</td>
</tr>
<tr>
<td>*4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>+P1</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td>P2</td>
</tr>
<tr>
<td>12</td>
<td>*13</td>
<td>14</td>
<td>15</td>
<td>+P3</td>
</tr>
</tbody>
</table>

**Writes To 4, 13 And Respective Parity Blocks**

- Disk 0 and Disk 1 can be accessed in parallel
- Disk 4 prevents any parallelism

**RAID-4 throughput under random small writes is** \( \frac{R}{2} \) **MB/s** *(terrible!)*
RAID Level 5: Rotating Parity

- Solves the small write problem
  - Rotate the parity blocks across drives
  - Remove the parity-disk bottleneck for RAID-4

<table>
<thead>
<tr>
<th>Disk 0</th>
<th>Disk 1</th>
<th>Disk 2</th>
<th>Disk 3</th>
<th>Disk 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>P0</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td></td>
<td>P1</td>
</tr>
<tr>
<td>10</td>
<td>11</td>
<td></td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td>P2</td>
<td>13</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>P3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>P4</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
</tr>
</tbody>
</table>

RAID-5 With Rotated Parity
RAID-5 Analysis

• **Capacity**
  • The useful capacity for a RAID group is \((N - 1)\)
  • Same as RAID-4

• **Reliability**
  • RAID-5 tolerates 1 disk failure and no more
  • Same as RAID-4

\(N\) : the number of disks
RAID-5 Analysis (Cont.)

**Performance**

- Sequential read and write
  - A single read and write request
  - Random read: a little better than RAID-4
    - RAID-5 can utilize all of the disks

- Random write: \( \frac{N}{4} \cdot R \) MB/s
  - The factor of four loss due to each write generating 4 total I/O operations

- Performs the same as or better than RAID-4 in all cases
  - RAID-5 has almost completely replaced RAID-4 in the marketplace

\( N \) : the number of disks
RAID Comparison: A Summary

N: the number of disks
T: the time that a request to a single disk takes
B: the blocks per disk

<table>
<thead>
<tr>
<th>RAID-0</th>
<th>RAID-1</th>
<th>RAID-4</th>
<th>RAID-5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Capacity</td>
<td>N \cdot B</td>
<td>(N \cdot B)/2</td>
<td>(N – 1) \cdot B</td>
</tr>
<tr>
<td>Reliability</td>
<td>0</td>
<td>1 (for sure)</td>
<td>1</td>
</tr>
<tr>
<td>Throughput</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sequential Read</td>
<td>N \cdot S</td>
<td>(N/2) \cdot S</td>
<td>(N-1) \cdot S</td>
</tr>
<tr>
<td>Sequential Write</td>
<td>N \cdot S</td>
<td>(N/2) \cdot S</td>
<td>(N-1) \cdot S</td>
</tr>
<tr>
<td>Random Read</td>
<td>N \cdot R</td>
<td>N \cdot R</td>
<td>(N-1) \cdot R</td>
</tr>
<tr>
<td>Random Write</td>
<td>N \cdot R</td>
<td>(N/2) \cdot R</td>
<td>\frac{1}{2} R</td>
</tr>
<tr>
<td>Latency</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Read</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>Write</td>
<td>T</td>
<td>T</td>
<td>2T</td>
</tr>
</tbody>
</table>

RAID Capacity, Reliability, and Performance
RAID Comparison: A Summary

• **Performance** and do not care about reliability → RAID-0 (Striping)

• **Random I/O** performance and **Reliability** → RAID-1 (Mirroring)

• **Capacity** and **Reliability** → RAID-5

• **Sequential I/O** and Maximize **Capacity** → RAID-5