Fast Wide Area Live Migration with a Low Overhead Through Page Cache Teleportation

Soramichi Akiyama*, Takahiro Hirofuchi†, Ryousei Takano†, Shinichi Honiden‡*

Email: akiyama@nii.ac.jp * University of Tokyo † National Institute of Advanced Industrial Science and Technology ‡ National Institute of Informatics

- Abstract -
Wide area live migration is essential for VM placement optimization between datacenters. However, it takes long time due to slow WAN and large page cache in the VM memory. Restorable page cache is redundant on the VM memory & disk image thus must be deduplicated. We propose page cache teleportation, which copies restorable page cache from the disk image instead of transferring it via WAN. It greatly reduces the total migration time of wide area migration with lower performance overhead.

- Challenges of WAN Migration -
1. Bandwidth is narrow between datacenters
   • links are shared by many users
   ➔ Reducing transferred data is important
2. Large page cache is duplicated on VM memory & disk
   • workloads with large data (e.g. web server, DB)
   • OSs assign all free memory for page cache
   ➔ page cache must be deduplicated to reduce the cost

Fast wide area live migration focusing on page cache is required!

- Page Cache Teleportation -

Kernel Module v.s. Alternative Approaches

<table>
<thead>
<tr>
<th></th>
<th>Kernel Module (our proposal)</th>
<th>Introspection-based</th>
<th>Disk I/O-Monitoring*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Implementation</td>
<td>Easy (&lt;200 loc)</td>
<td>Hard</td>
<td>Middle</td>
</tr>
<tr>
<td>Runtime overhead</td>
<td>None</td>
<td>None</td>
<td>Disk write hooked</td>
</tr>
<tr>
<td>Migration overhead</td>
<td>Small (&lt;1 sec)</td>
<td>Big (binary scan)</td>
<td>Small</td>
</tr>
<tr>
<td>Guest modification</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
</tbody>
</table>

*Jo et al., VEE’13

- Experimental Results -
Total migration time in the original/proposal with 1G mem/1 vCPU VM Web Server:
• Simulated static web site
• HTMLs are 300 MB in total

TPC-C:
• Simulated net shopping site
• DB data is 1.9GB in total

<table>
<thead>
<tr>
<th></th>
<th>Original</th>
<th>Proposed</th>
<th>Original</th>
<th>Proposed</th>
<th>Original</th>
<th>Proposed</th>
<th>Original</th>
<th>Proposed</th>
<th>Original</th>
<th>Proposed</th>
</tr>
</thead>
<tbody>
<tr>
<td>File read throughput (blocks/s)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0</td>
<td>50</td>
<td>100</td>
<td>500</td>
<td>1000</td>
<td>5000</td>
<td>10000</td>
<td>50000</td>
<td>100000</td>
<td>500000</td>
</tr>
</tbody>
</table>

With our proposal it recovers in 3 seconds (top). If however, page cache is deleted to reduce transferred data, it degrades for 15 seconds (bottom).