Techniques Applied for Memory Management in modern OS

Introduction

We have covered the basics steps to virtualize the CPU and the main approaches to virtualize memory. Using paging to compartmentalize memory in pages keeping track of the VPN and the VFN has shown great performance and stability when accessing data, we briefly covered how the programmer can implement techniques and algorithms to help with the cons that paging brings as well as some techniques to swap memory to disk.

Suppose that a system has a 32-bit (4GB) virtual address space. It has 1GB of physical memory, and uses 1MB pages.

1. How many virtual pages are there in the address space?

   There are 2

2. How many physical pages are there in the address space?

   There are 4096

3. How many bits are there in the offset?

   There are 1024

4. How many bits are there in the virtual page number?

   There are 12

5. How many bits are there in the physical page number?

   There are 10

6. Some entries of the page table are shown below (all values are in hex, and all entries shown are valid). Translate virtual address 0x410423 to a physical address, using the translations in this page table.

   Translation: 0xDD10423

Entry	Number
Value	
0	1F
1	3C
2	55
3	9C
4	DD
5	EE
6	99
...	...
20	2F
21	4C
22	65
23	AC
24	ED
25	FE
26	100
...	...
40	11F
41	13C
42	155
43	19C
44	1DD
45	1EE
46	199
...	.

Conclusions

Even though the assignment did not covered all the important aspects involved in memory mangement and the possible solutions a OS programmer can implement, it did force me to review carefully how memory is allocated using page tables and memory swaping.

References:

-Arpaci-Dusseau, R. & Arpaci-Dusseau, A. (2012). Operating Systems: Three Easy Pieces. Madison, WI: University of Wisconsin-Madison. Available at http://pages.cs.wisc.edu/~remzi/OSTEP/