Skip to main content

CSSE2310

Home / Virtual Memory / Page Tables

Page Tables

Submitted by nhoy on Tue, 08/30/2016 - 22:23

Find memory required for page table

To find the memory required for a page table, follow the following steps.

1. work out what pages (or ranges of pages) you need to map.
2. a page is mapped by a page table entry in some page table, so for each page work out which page table it is mapped in.
3. make note of each unique page table. For a multilevel page table with two levels, the root page table is always required (1 page) plus all the page tables from the previous step.

$$\text{memory} = \text{number of page tables} \cdot \text{page size}$$

Example

Given a system with

- 32-bit virtual addresses,
- 4kB pages,
- page table entries are 4 bytes, and
- it uses two level page table.

How much memory is required for the page table of a process which uses the following memory: 122880 bytes starting at 1024 and 30719 starting at 125825024

So we have a multilevel page table with two levels. That means we have a single root level-1 page table, each of whose entries point to a level-2 page table. The entries of each level-2 page table are the actual page table entries that map the virtual pages to physical page frames.

First we want to know how many entries each page has

$$\frac{\text{page size}}{\text{entry size}} = \frac{4\text{kB}}{4\text{B}} = 1024 \text{ entries per page table.}$$

We have the first region starting at address 1024, but what page is that on? Each page has 4096 bytes (0-4095), so it must be on the first page (page 0). Here's the formula, note that we use integer division (round down).

$$\frac{\text{address}}{1024}$$

$$\text{page} = \frac{}{\text{page size}} = \frac{}{4096} = 0$$

What page does this memory region end on? It starts at address 1024 and is 122880 bytes long, so it ends at address $1024 + 122880 = 123904$. Plug that into the page formula, and we get page $123904/4096 = 30$.

Address	Page	L2 Page Table
1024	0	0
123904	30	0

The region starts on page 0 and ends at page 30. Where are the page table entries for these pages? The first pointer in the root of the multilevel page table points to a level-2 page table that maps the first 1024 *pages* (0 - 1023). So entries for pages 0-30 are in that first level-2 page table.

What about the second memory region? It starts at 125825024 and is 30719 bytes long.

125825024 is on page 30719 and the region ends at address $125825024 + 30719 = 125855743$, which is on page 30726.

Address	Page	L2 Page Table
125825024	30719	29
125855743	30726	30

Where are the page table entries for this region? The first level-2 page table maps pages 0-1023, the second maps 1024-2047 etc. We have the formula

$$\text{L2 page table} = \frac{\text{page}}{\text{entries per page table}}$$

So the #29 level-2 page table contains the entry for the start page (30719) and the #30 level-2 page table contains the entry for the end page (30726).

For the first memory region, we needed a single level-2 page table and for the second region we needed two level-2 page tables.

Root page table

Entry	Maps to
0	address of 1st level-2 page table
...	...

29	address of #29 level-2 page table
30	address of #30 level-2 page table
...	...

So, we need the root table (4kB) plus the three level-2 page tables (4kB each). In total, we need $4 \cdot 4\text{kB} = 16\text{kB}$ for our multilevel page table.

‹ Memory Accesses

up

Printer-friendly version

Search form

Search