

Anu Khatri

CSCE 410-500

Professor Da Silva

5 April 2021

P4 – Design Document

The first change that needs to happen will be in the constructor.

```
unsigned long *page_table = (unsigned long *) (process_mem_pool->get_frames(1) * PAGE_SIZE);
```

The page table will need to have a frame from the process_mem_pool instead of kernel_mem_pool in order for the recursive lookup to work.

```
for (i = 1; i < 1023; i++){
    page_directory[i] = 0 | 2;
}

// Assigning last index in page_directory back to the page directory
page_directory[1023] = (unsigned long) page_directory | 3;
```

Here, the last index of the page_directory will point back to the beginning of the page directory so that we are able to use the recursive trick.

In handle_fault, there are many more changes that need to be made.

```
unsigned long page_table_address = process_mem_pool->get_frames(1) * PAGE_SIZE;
```

Firstly, the page_table_address is using the process_mem_pool now instead of the kernel_mem_pool.

```
for (i = 0; i < 1024; i++){
    unsigned long address;
    unsigned long p1_section = 0xFFC00000;
    unsigned long pd_section = page_dir_index << SHIFT_12;
    unsigned long i_section = i << SHIFT_2;
    address = p1_section + pd_section + i_section;
    trick_address = (unsigned long*) address;
    *trick_address = 2;
}
```

In the for loop that deals with the inner page table, this needs to be modified to accommodate for the trick address. I divided the bit shifting and such into different sections so that it would not be too complicated. The address is comprised of p1_section (the 10 1s at the beginning to signify the 1023 value), pd_section (the page directory index), and the i_section (the value of i in the for loop).

```

unsigned long address2;
unsigned long p1_section = 0xFFC00000;
unsigned long pd_section = page_dir_index << SHIFT_12;
unsigned long pt_section = page_table_index << SHIFT_2;
address2 = p1_section + pd_section + pt_section;
trick_address = (unsigned long *) address2;

page_table2 = (unsigned long *) (page_dir[page_dir_index] & 0xFFFFF000);
*trick_address = (process_mem_pool->get_frames(1) * PAGE_SIZE) | 3;

```

The next thing that needs to be changed is in my else statement (which deals with when

`page_dir[page_dir_index] & 1`

is true. Here, the trick address is a little different in that instead of the `i` getting used earlier, this one will have the page table index in there, so the construction is a little different (I have `pt_section` instead of `i_section`). There is still some similar bitshifting involved.