

Lab11 Report

12011702 张镇涛

1

一个进程有一个 `mm_struct`。

`mm_struct` 是内存描述符，描述一个进程的整个虚拟地址空间，包含装入的可执行映像信息以及进程的页目录指针 `pgdir` 等。

2

`vma_struct` 结构体描述一段连续的虚拟地址，从 `vm_start` 到 `vm_end`，描述了虚拟地址空间的一个区间(简称虚拟区)。

每个页表（每个虚拟地址空间）可能包含多个 `vma_struct`，也就是多个访问权限可能不同的，不相交的连续地址区间。

3

在CPU访问虚拟地址时，找不到其对应的物理内存的时候触发缺页中断。

以下情况可能导致page fault被触发：

1. 页表中没有虚拟地址对应的PTE（虚拟地址无效 或 虚拟地址有效但没有分配物理内存页）
2. 现有权限无法操作对应的PTE

4

major page fault指的是访问的虚拟地址内容不在内存中，需要从外设载入。其处理方法是将交换区中的页面重新载入内存并建立虚拟地址和物理地址的映射关系。

其在实验代码中对应 `kern/mm/vmm.c` 中 `do_pgfault()` 函数中：

```
344     } else {  
345         if (swap_init_ok) {  
346             struct Page *page = NULL;  
347             swap_in(mm, addr, &page); //According to the mm AND addr, try  
348                                     //to load the content of right disk page  
349                                     //into the memory which page managed.  
350             page_insert(mm->pgdir, page, addr, perm); //According to the mm,  
351                                                         //addr AND page, setup the  
352                                                         //map of phy addr <-->  
353                                                         //logical addr  
354             swap_map_swappable(mm, addr, page,  
355                                1); //make the page swappable.  
356             page->pra_vaddr = addr;  
357         }
```