OSLab2 文档

陈俊 5140379064

1. Physical Page Management

这一部分主要是按照完成 boot_alloc，mem_init 以及 page 部分的相关函数。

boot_malloc 主要是用于在未初始化 page 时暂时分配一些空间。由提示可知，end 位于 kernel 的 bss 段之后，也就是整个 kernel 文档之后。在当前状态，end 之后的空间都是空闲和可用的，因此，我们可以分配这些空间使用。在 boot_malloc 函数中，我们线性按序分配之后的地址，并用 nextfree 来记录下一个未分配的地址，并将未更新前的 nextfree 作为返回值，以该地址开头 size 长度的空间已经被分配，供之后使用。同时，每次需要 ROUNDUP 一下，以保证对齐。由于考虑到在最开始的物理地址映射中，只映射了 4M，因此，当分配超过 4M 时，认为 out of memory，并报错。

mem_init 用 boot_alloc 分配并初始化了 npages 个 Page，用于记载各个页的信息（ref 及下个空白页），同时，用 boot_map_region 函数，映射了多个虚拟地址和物理地址的映射关系（UPage，bootstack 及 kernelbase）。

Page_init 初始化了各个页的信息。将第一个页、IOHOLE 以及 kernelbase 到 boot_alloc(0)作为已分配的页，而剩下的页设为 free，并修改它们的 ref 及 link。（通过询问助教，由于之后 lab 会重新修改 gdt，因此 boot 中的 gdt 部分置为 free，可以被覆盖）

Page_alloc 主要用于空闲页的分配，原理很简单，就是从空闲页链表中取出一个空闲页，将其地址返回即可（若需置为 0，根据 page2kva 找到页，用 memset 将内容置为 0）。

Page_free 用于页的 free，只需将页放回空闲页链表即可。

2. Virtual Memory

这一部分主要实现五个函数，分别为：pgdir_walk，boot_map_region，page_lookup，page_remove 以及 page_insert。

Pgdir_walk 用于返回二级页表中 vaddr 对应的 PTE 的指针（若为大页，则返回第一级页表（都是 PTE）的对应指针）。由于页地址的分配为 10 10 12 或 10 22，因此只需通过数组操作返回对应的指针即可。如果一级页表下该虚拟地址的二级页表不存在，则用 alloc_page 分配一个新页（create 为 1 的情况），并且通过 page2pa 获得物理地址，并与 perm 一起填写 pde，再返回二级页表中对应 pte 的指针。

Boot_map_region 主要用于物理地址段与虚拟地址段的映射。根据 size，隔 PGSIZE 累加虚拟地址，通过 pgdir_walk 函数获得 pte 的指针，填入对应物理地址或上 perm 和 PTE_P，以完成实与虚的映射。

Page_lookup 用于返回虚拟地址对应页的指针，只需调用 pgdir_walk，用 PTE_ADDR 获得其中的物理地址，并且用 pa2page 转化获得页的地址即可。同时，也将 pte 的指针传入 pte_store，作为其的值。

Page_insert 用于将物理页对应到某个虚拟地址，若原先虚拟地址有映射，则取消之前的映射，与新的页进行新的映射。也是利用 pgdir_walk，看原先是否有映射，若有，则调用 page_remove，之后再修改 pte 的内容以修改页的对应关系。

Page_remove 用于取消页与虚拟地址的映射。通过 page_lookup 找到对应页的地址，调用 page_decref，同时调用 tlb_invalidate，消除 tlb 中的对应内容。

3. Kernel Address Space

这一部分主要实现了大页的映射。（mem_init 中的 boot_map_region 在第一部分实现了）。这里需要增加 boot_map_region_large 函数，用于对应大页。由于 perm 中并未有 PTE_PS，因此我专门写了 pgdir_walk_large，用于大页页表 pte 的查询。这个比 4K 的简单，由于是 10 22，因此只需一级页表，直接返回对应 pte 的地址即可。同时，在累加的时候，也要讲 PGSIZE 改为 PTSIZE。这里，我直接将 mem_init 中 kernelbase 映射部分改为了 boot_map_region_large，并且，在之前开启了 cr4 中的 CR4_PSE，表示开启大页。

4. Challenge

Chanllege 部分我选择了第一个来进行实现。主要实现了 showmappings，changeperm 以及 dump 的功能。具体已在 answer-lab2.txt 中介绍，在此不作赘述。