CHINA LINUX KERNEL
中国Linux内核开发者大会

華中科技大學
网络空间安全学院
School of Cyber Science and Engineering, HUST

第19届中国Linux内核开发者大会

赞助单位

華中科技大學 | HUAWEI | 阿里云 | FUJITSU | 腾讯云
oppo | 火山引擎 | vivo | 龙芯中科 LOONGSON TECHNOLOGY | 蚂蚁集团 ANT GROUP

支持单位

清华大学 Tsinghua University | 迪捷软件 DIGIPROTO | OpenEuler | OpenAnolis 龙蜥社区 | OpenCloudOS

支持社区&媒体

CSDN | 云巅论剑 | sf 思否 | 阅码场 yomocode | InfoQ 极客传媒
51CTO | 开源江湖

2024年10月　湖北 • 武汉

華中科技大學

稀疏文件零页填充

白铠豪　阿里云内核开发工程师

目录

01 背景 & 问题

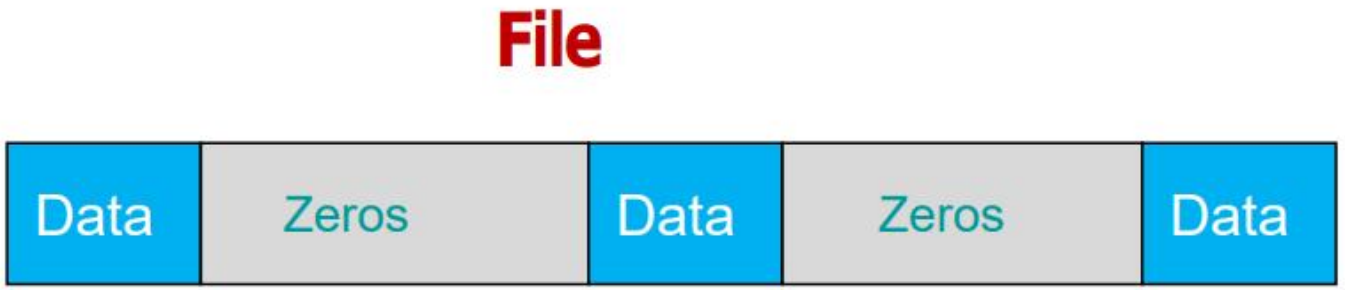
背景

- 稀疏文件（Sparse file）是一种文件存储格式，它可以高效地存储那些包含大量空白区域（即零字节区域）的数据。

优点：**可以创建实际大小远大于物理存储容量的文件**

缺点：并非所有的文件系统都支持稀疏文件；不支持稀疏文件的文件系统可能会将其作为普通文件处理，从而消耗大量磁盘空间。

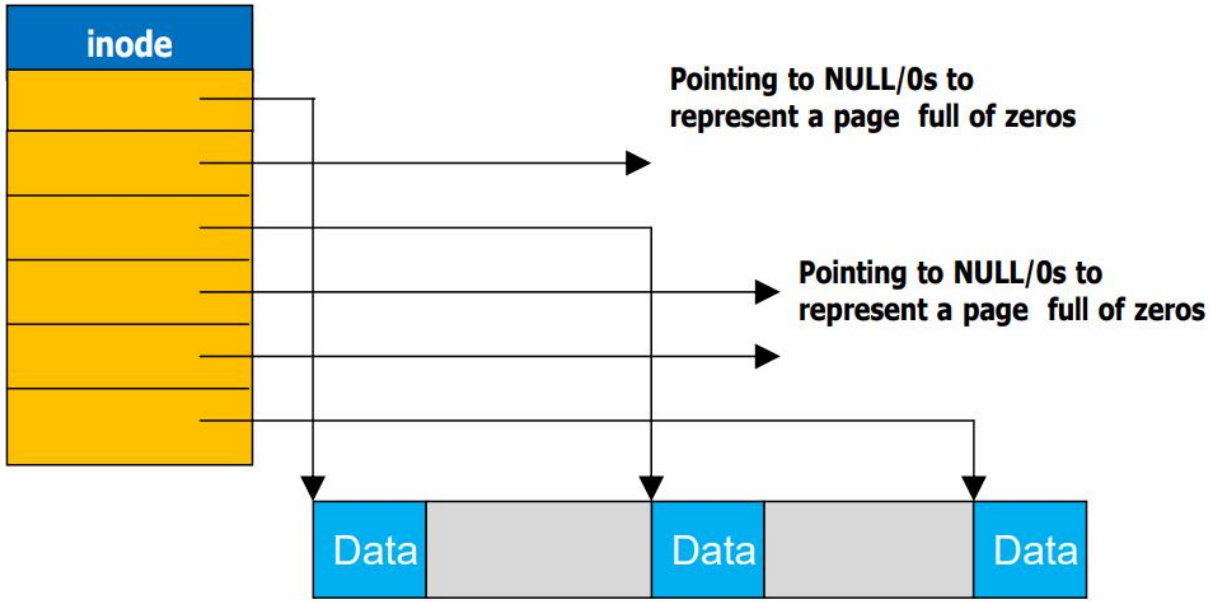
系统零页（Zero Page）通常指的是一个被初始化为全零的内存页，一般用于未初始化匿名页的分配。



稀疏文件

问题

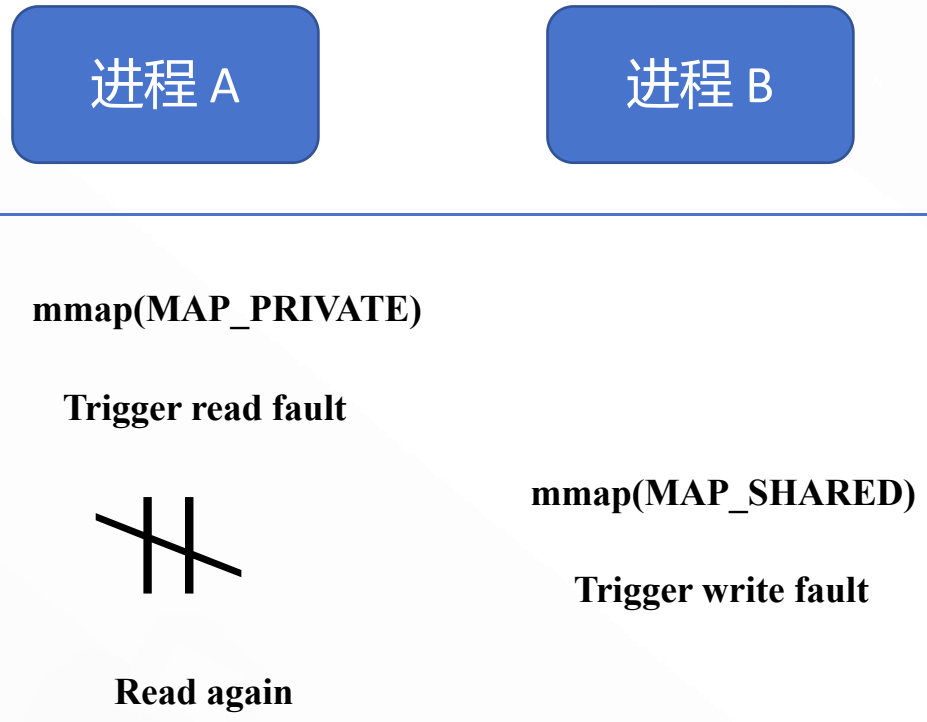
- 模板技术将模板容器的内存快照保存为稀疏文件，在启动新的实例时直接使用文件映射。使用模版技术可以**大幅减少容器启动延时，提高并发度，**而模板技术在保存内存快照时类似于QEMU的RAW格式，**仅有极少部分有效数据**，全零部分在运行时会被匿名页重映射。
- 稀疏文件中文件位移量可以大于文件的当前长度。位于文件中但没有写过的字节都被设为 0（称为hole）。在对文件Hole进行读写时会触发页分配并添加进Page Cache中。如果为文件的私有映射，**Page cache会增加一些无意义的全零page。**



映射方式

02 设计 & 实现

待解决的问题



- **MAP_PRIVATE/MAP_SHARED页同步**

  由于MAP_SHARED/MAP_PRIVATE映射的VMA间需要保持数据同步，因此两者操作的内存页需要保证同步，不允许出现系统零页填充的同时新增page cache，在unmap的情况下依然填充了系统零页造成数据丢失等同步问题。

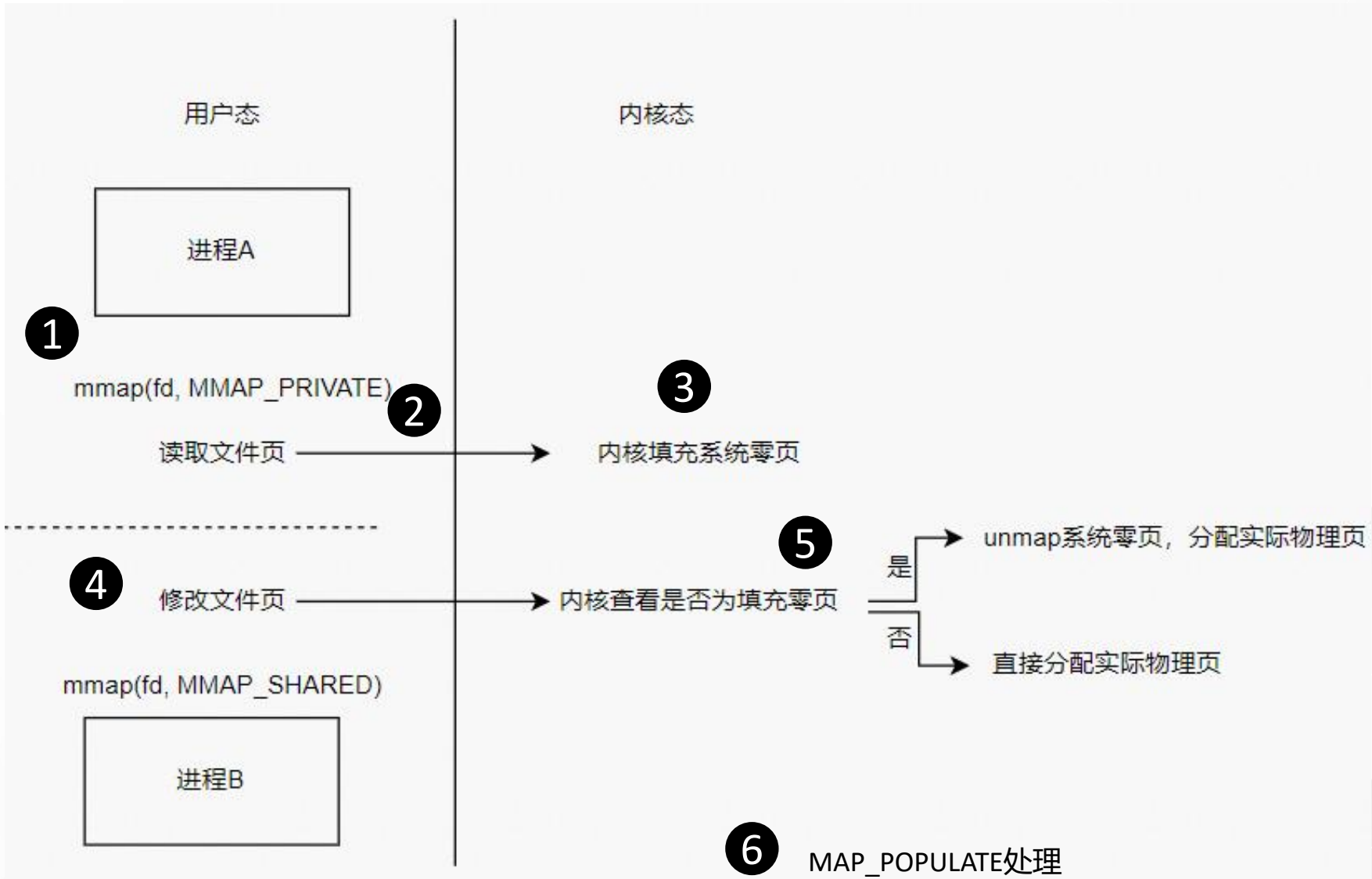
- **语义一致性**

  对于系统零页来说，需要保证其不被原有文件页Page Fault流程污染，还需要保证进程计数等符合预期。

- **动态开关设计**

  动态开关设计需要考虑开启/关闭的开销，避免对原有性能的影响。

方案设计


用户态
内核态
进程A
1
mmap(fd, MMAP_PRIVATE)
2
读取文件页
3
内核填充系统零页
4
修改文件页
5
内核查看是否为填充零页
是
unmap系统零页，分配实际物理页
否
直接分配实际物理页
mmap(fd, MMAP_SHARED)
进程B
6
MAP_POPULATE处理

竞态分析

MMAP_PRIVATE	MMAP_SHARED
do_read_fault	
	do_shared_fault
check pagecache	
	alloc_page
	add_to_pagecache
	try_to_unmap_zeropage
set_pte	

03 收益

收益

- 测试用例

```
total 32G
32G ---Sr-s--T 1 root root 32G Aug 10 10:56 mem_16C_32768M
```

```
total 576M
576M -rw-r--r-- 1 root root  32G Aug  6 10:52 mem_16C_32768M
```

  - 在提高了容器冷启动并发度的同时，节省了host Page Cache的内存占用。

Q & A