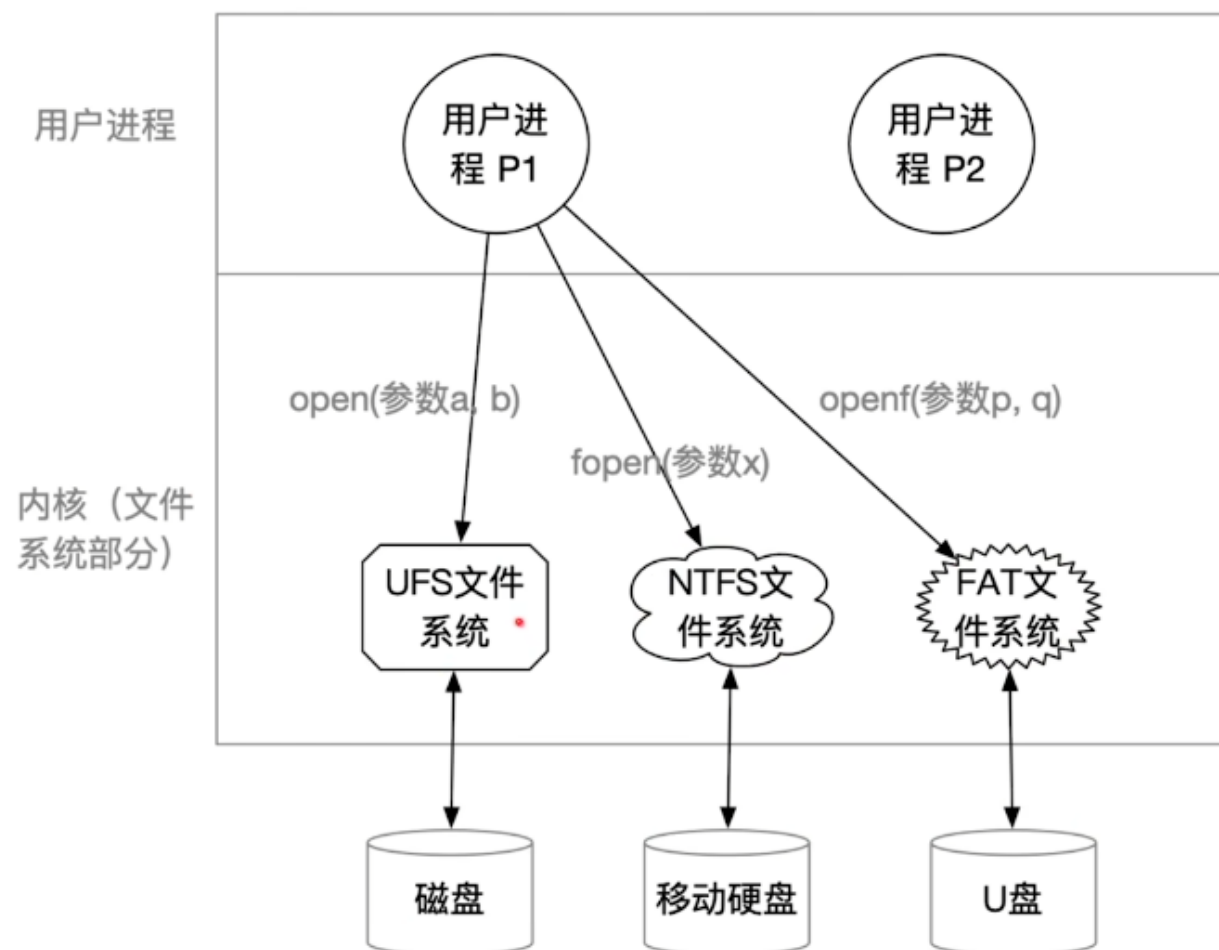
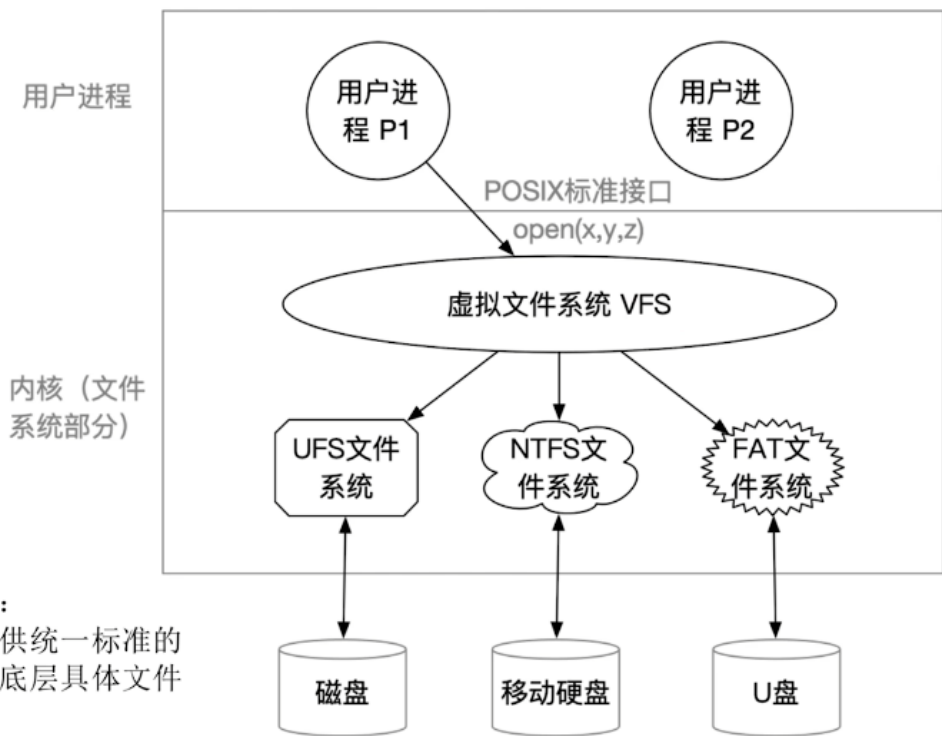


# 虚拟文件系统

## 普通的文件系统

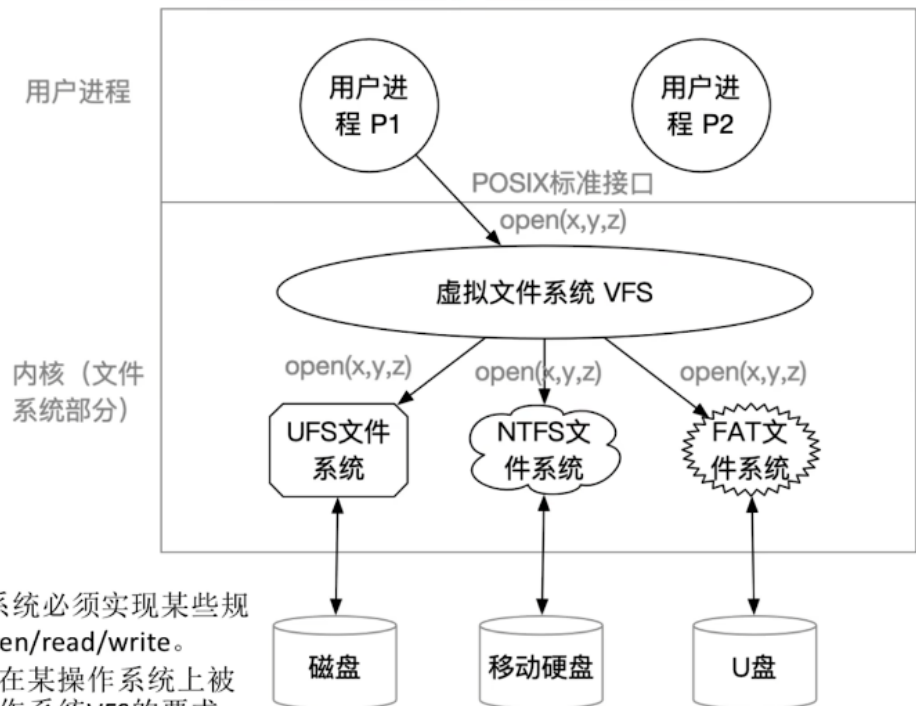


## 虚拟文件系统



虚拟文件系统的特点：

①向上层用户进程提供统一标准的系统调用接口，屏蔽底层具体文件系统的实现差异



虚拟文件系统的特点：

②VFS要求下层的文件系统必须实现某些规定的函数功能，如：open/read/write。  
一个新的文件系统想要在某操作系统上使用，就必须满足该操作系统VFS的要求

存在的问题：不同的文件系统，表示文件数据结构各不相同。打开文件后，其在内存中的表示就不同

UFS文件系统的目录项

文件名	i结点号
-----	------

所有者	
创建时间	
总块数	1
共享计数	
直接索引0	5
直接索引1	
直接索引2	
直接索引3	
直接索引4	
直接索引5	
直接索引6	
直接索引7	
直接索引8	
直接索引9	
一级间接	
二级间接	
三级间接	

索引结点

FAT文件系统的目录项

文件名	文件类型	...其他...	文件大小	起始块号
-----	------	----------	------	------

文件名
文件大小
创建者
文件格式
.....
函数功能指针

vnode (v结点)

UFS文件系统的目录项

文件名	i结点号
-----	------

所有者	
创建时间	
总块数	1
共享计数	
直接索引0	5
直接索引1	
直接索引2	
直接索引3	
直接索引4	
直接索引5	
直接索引6	
直接索引7	
直接索引8	
直接索引9	
一级间接	
二级间接	
三级间接	

inode

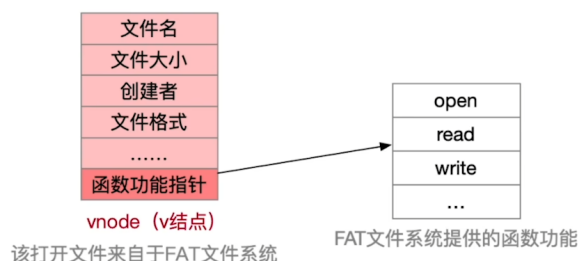
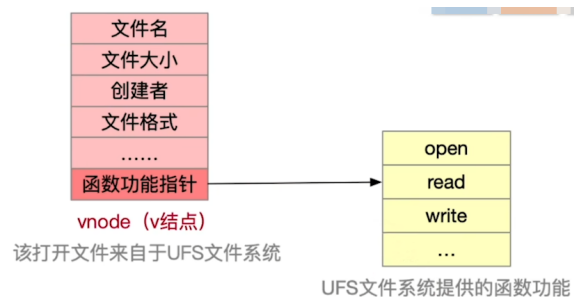
虚拟文件系统的特点：

③每打开一个文件，VFS就在主存中新建一个vnode，用统一的数据结构表示文件，无论该文件存储在哪个文件系统。

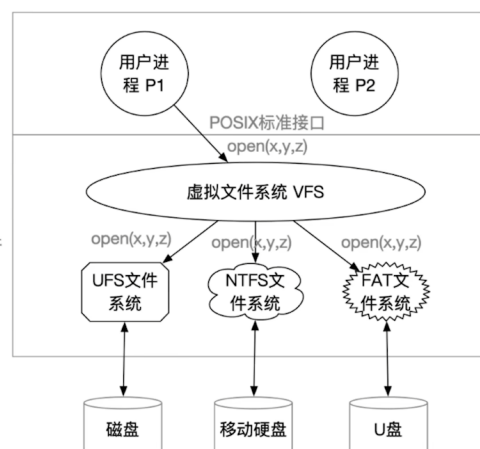
FAT文件系统的目录项

文件名	文件类型	...其他...	文件大小	起始块号
-----	------	----------	------	------

注意：vnode只存在于主存中，而inode既会被调入主存，也会在外存中存储。



用户进程



打开文件后，创建vnode，并将文件信息复制到vnode中，vnode的功能指针指向具体文件系统的函数功能。