第9章 Linux文件系统

北京理工大学计算机学院

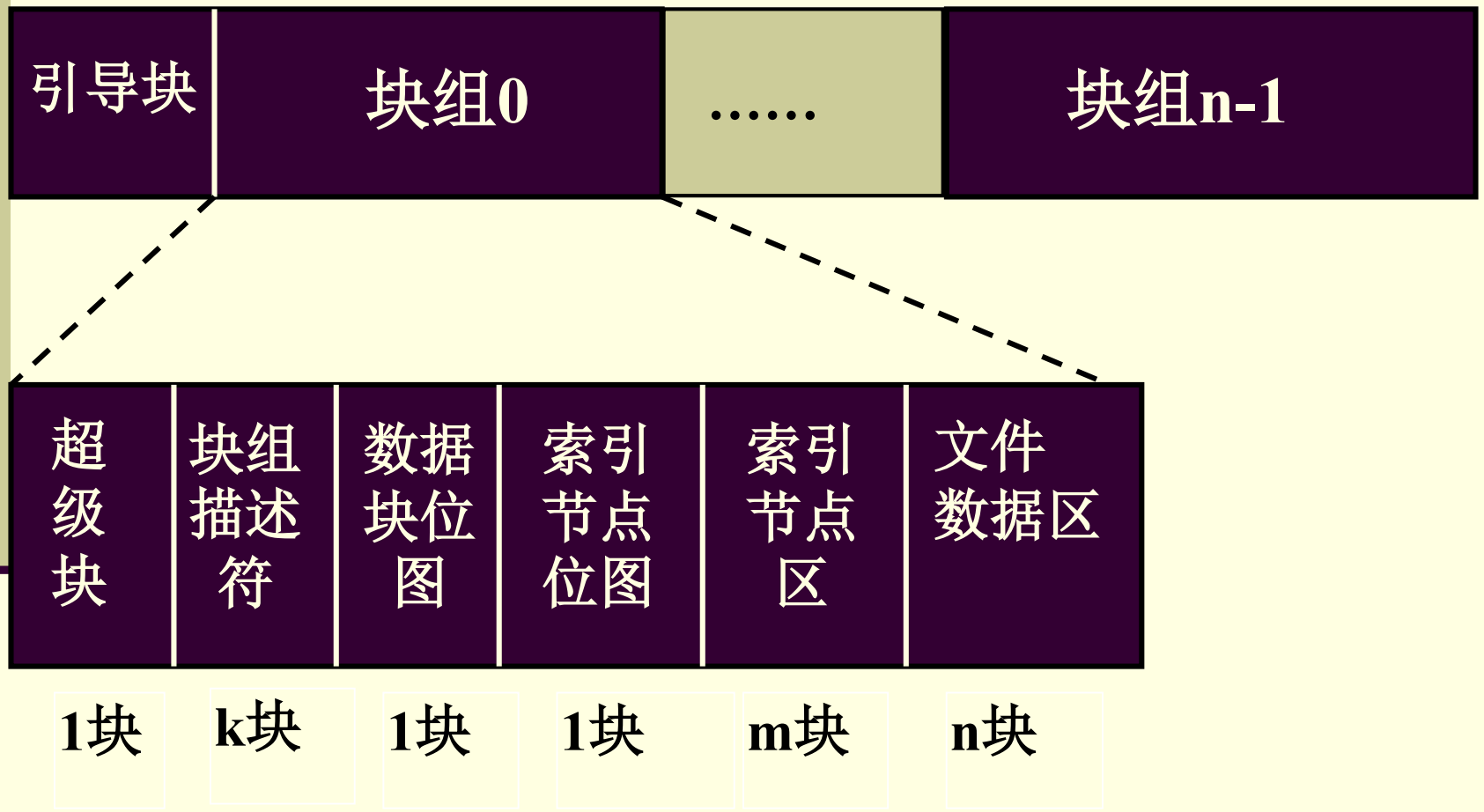
9.1 Ext2磁盘涉及的数据结构

- Linux最初采用的是Minix的文件系统，但是Minix只是一种试验性的操作系统，其文件系统的大小仅限于64M字节。当Linux成熟时，引入了扩展文件系统（ Extended Filesystem, Ext FS），但提供的性能不令人满意。在1994年引入了第二扩展文件系统（Ext2），它相当高效和强健，已成为广泛使用的Linux文件系统。

9.1.1 Ext2文件卷的布局

- Ext2把磁盘块分为组，每组包含存放在相邻磁道的数据块和索引节点。块组的大小相等并顺序安排。
- Ext2用“块组描述符”来描述这些块组本身的结构信息，同时将超级块和所有的块组描述符重复存储于每个块组中。
- Ext2通过“位图”来管理每个块组中的磁盘块和索引节点。盘块位图，索引节点位图。

Ext2示意图


引导块
块组0
……
块组n-1
超级块
块组描述符
数据块位图
索引节点位图
索引节点区
文件数据区
1块
k块
1块
1块
m块
n块

- 超级块存放整个文件卷的资源管理信息。
- 索引节点存放文件的管理控制信息。
- 只有块组0中所包含的超级块和组描述符才由内核使用，而其余的超级块和组描述符保持不变，事实上，内核甚至不考虑它们。当系统对文件系统的状态执行一致性检查时，就引用存放在块组0中的超级块和组描述符，然后把它们拷贝到其它所有的块组中。

- 块组的多少取决于分区的大小和块的大小。盘块位图必须存放在一个单独的块中（盘块位图的每1位对应着块组中的一个盘块，1表示已分配，0表示空闲）。
- 考虑一个8GB的分区，盘块的大小为4KB。每个4KB的盘块位图可描述32K个磁盘块，即128MB。因此，最多需要64个块组。

9.1.2 超级块

```
struct  ext2_super_block {
__le32  s_inodes_count;          索引节点的总数
__le32  s_blocks_count;          盘块的总数
__le32  s_free_blocks_count;     空闲块计数
__le32  s_free_inodes_count;     空闲索引节点数
__le32  s_log_block_size;        盘块的大小
__le32  s_blocks_per_group;      每组中的盘块数
__le32  s_inodes_per_group;      每组索引节点数
__le16  s_inode_size; 磁盘上索引节点结构的大小
……};
```

- s_log_block_size字段以2的幂次方表示盘块的大小，且以1024B为单位。幂指数为0表示盘块大小为1024B，为1表示盘块大小为2048B等等。
- 假设每个块组包含4096个索引节点，若想确定索引节点13021在磁盘上的地址。

  13021/4096=3……733

  该索引节点在第3个块组的索引节点表的第733个表项中。

9.1.3 块组描述符

```
struct  ext2_group_desc {
__le32  bg_block_bitmap;      盘块位图的块号
__le32  bg_inode_bitmap;      索引节点位图的块号
__le32  bg_inode_table; 索引节点表的第一个盘块
   块号
__le16  bg_free_blocks_count;组中空闲块的个数
__le16  bg_free_inodes_count;组中空闲索引节点
   的个数
__le16  bg_used_dirs_count; 组中目录的个数
……};
```

9.1.4 文件目录与索引节点结构

- 把通常的文件目录项分成简单目录项和索引节点两部分。
- 简单目录项包含了文件名和索引节点号等，可以提高文件目录的检索速度。
- 系统只保留一个索引节点，就可实现多条路径共享文件，减少信息冗余。

目录项结构

```
struct ext2_dir_entry_2{
__le32  inode;      索引节点号
__le16  rec_len;    目录项长度
__u8   name_len;    实际文件名长度
__u8  file_type;    文件类型
char  name[255];    文件名，变长数组
}
```

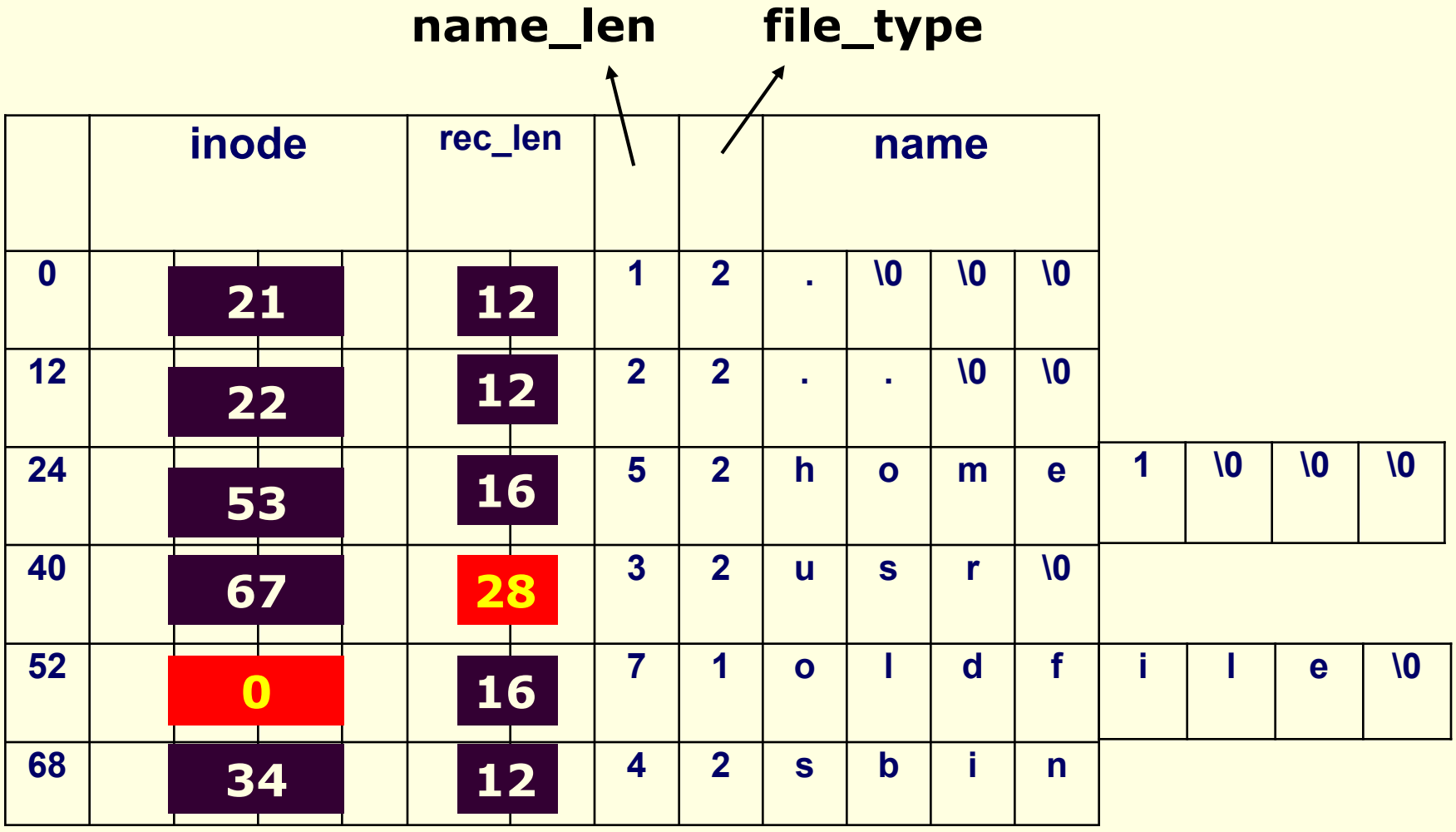
文件类型

- 0：不可知
- 1：普通文件
- 2：目录文件
- 3：字符设备文件
- 4：块设备文件
- 5：有名管道
- 6：套接字文件
- 7：符号链接

- name字段存放文件的名字，最多为255B，实际字符数为name_len。
- 文件名必须是4B的整数倍。
- 目录结构是变长的

目录项

28=12+16


name_len
file_type
inode
rec_len
name
0
21
12
1
2
.
\0
\0
\0
12
22
12
2
2
.
.
\0
\0
24
53
16
5
2
h
o
m
e
1
\0
\0
\0
40
67
28
3
2
u
s
r
\0
52
0
16
7
1
o
l
d
f
i
l
e
\0
68
34
12
4
2
s
b
i
n

索引节点（128B）

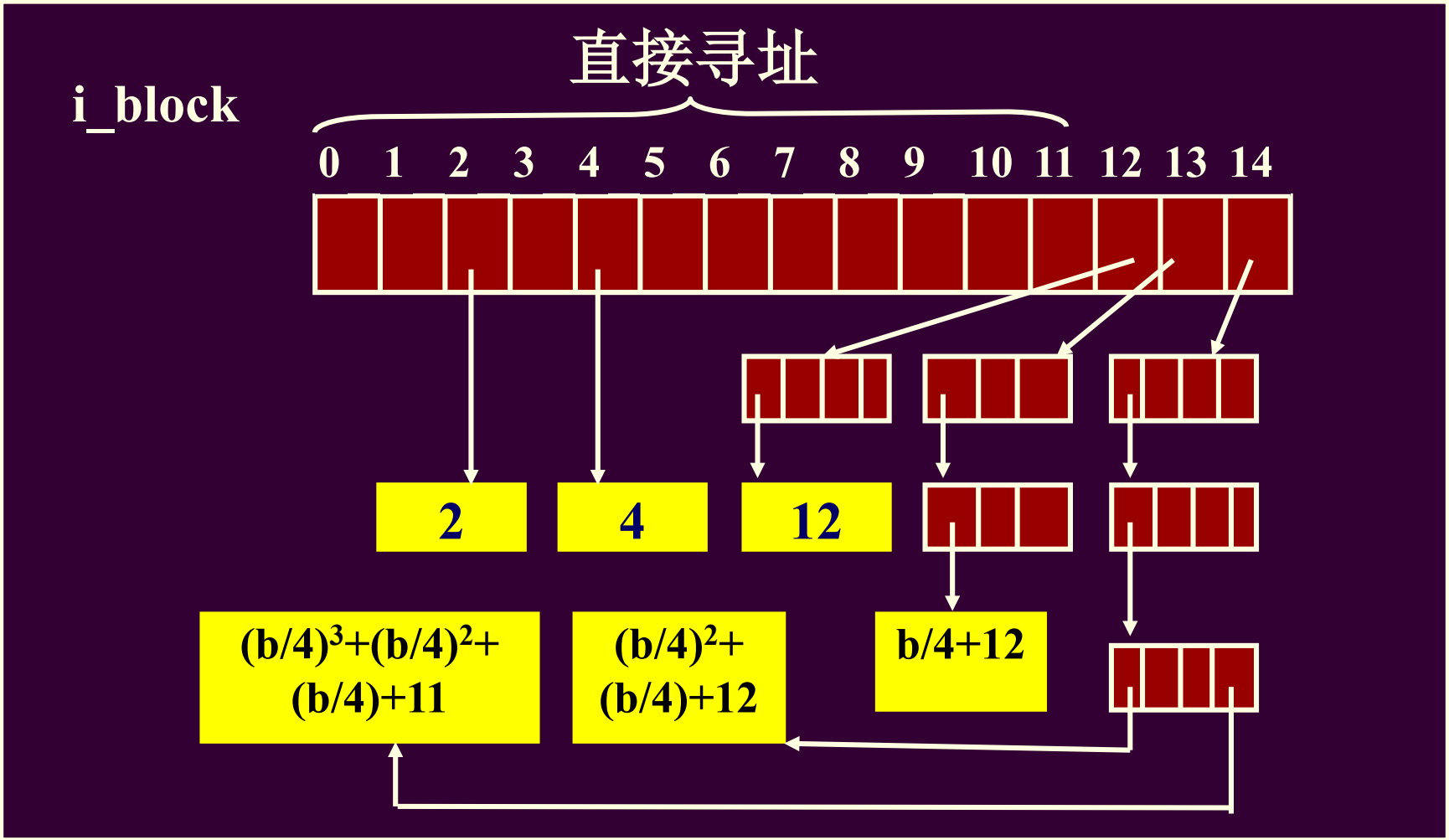
```
struct  ext2_inode {
__le16  i_mode;     文件类型和访问权限
__le16  i_uid;      拥有者的标识符
__le32  i_size;     以字节为单位的文件长度
__le16  i_gid;      组标识符
__le16  i_links_count;  硬连接计数
__le32  i_block[15];    索引表，15×4B
__le32  i_blocks;       文件的数据块数
__le32  i_file_acl;     文件访问控制表ACL
……};
```

i_mode

- 为0表示索引节点空闲
- 0～8位表示“拥有者、同组用户、其他用户”的RWX的访问权限。
- 12～15位表示文件类型，同目录结构的文件类型。
- i_links_count是文件的硬链接数。硬链接是指在同一个文件系统中，一个信息文件可能链接的文件名。（即一个索引节点与几个文件目录项相连）

索引表

- 索引节点表：由一些连续块组成，其中第一个块的块号存放在块组描述符的bg_inode_table字段中。每个索引节点的大小为128字节。索引节点的长度必须是2的幂。
- 索引表：i_block字段是一个有15个元素的数组，每个元素占4B，共60B。存放文件逻辑块号与相应物理块号之间的映射关系。数组的15个元素有4种类型。


直接寻址
i_block
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14
2
4
12
(b/4)³+(b/4)²+(b/4)+11
(b/4)²+(b/4)+12
b/4+12

索引表

索引表

- 最初的12个元素是直接索引项，给出文件最初的12个逻辑块号对应的物理块号。
- 索引12是一次间接索引块，是一个存放盘块号的一维数组。对应的文件逻辑块号从12到（b/4）+11，b是盘块大小，每个逻辑块号占4B。
- 索引13是二次间接索引块，对应的文件逻辑块号从b/4+12到 $(b/4)^2+(b/4)+11$。
- 索引14是三次间接索引块，对应的文件逻辑块号从 $(b/4)^2+(b/4)+12$到$(b/4)^3+(b/4)^2+(b/4)+11$。

符号链接文件

- 当符号连接文件的路径名小于60个字符时，就存放在i_block[ ]中；当大于60时，需要一个单独的数据块。
- 与硬链接区别：它不与文件的索引节点建立链接，可以跨文件系统（当为一个文件建立符号链接时，索引节点的硬链接计数不改变）。
- 设备文件、管道文件、套接字文件：信息存放在索引节点中，无需数据块。

9.2 Ext2的主存数据结构

- 为了提高效率，当安装Ext2文件系统时，存放在Ext2文件卷的数据结构中的大部分信息被复制到RAM中，从而避免了后来的内核多次读盘操作。
- 频繁更新的数据总要缓存在高速缓存中。内核可让其引用计数大于0来达到此目的。

Ext2数据结构的内存映像

- 超级块：
  - 磁盘结构：ext2_super_block
  - 内存结构：ext2_sb_info
- 块组描述符：
  - 磁盘结构：ext2_group_desc
  - 内存结构：ext2_group_desc
- 索引节点：
  - 磁盘结构：ext2_inode
  - 内存结构：ext2_inode_info

9.2.1 超级块和索引节点

- Ext2文件系统的磁盘超级块结构：ext2_super_block
- Ext2文件系统的内存超级块结构：ext2_sb_info
- ext2_sb_info包含了磁盘超级块的大部分内容。

ext2_sb_info 内存超级块

```
struct ext2_sb_info {
    unsigned long s_blocks_per_group;
    unsigned long s_inodes_per_group;
    unsigned long s_groups_count;
    struct ext2_super_block *s_es; /*指向ext2磁盘超级块所在的缓冲区*/
    ……};
```

ext2_inode_info

- 磁盘索引节点结构：ext2_inode。
- 内存索引节点结构：ext2_inode_info。

ext2_inode_info

```
struct  ext2_inode_info {
    __u32  i_data[15];  /*文件的索引表*/
    __u32  i_block_group;  /*索引节点所在块组的块组索引*/
    struct inode vfs_inode; /*索引节点对象*/
    ……};
```

9.3 磁盘空间管理

- 磁盘块和索引节点的分配和回收
- 文件的数据块和其索引节点尽量在同一个块组中。
- 文件和它的目录项尽量在同一个块组中。
- 父目录和子目录尽量在同一个块组中。
- 每个文件的数据块尽量连续存放。

9.4 Ext2提供的文件操作

VFS的文件操作	Ext2的方法
lseek	ext2_file_lseek( )
read	generic_file_read( )
write	generic_file_write( )
ioctl	ext2_ioctl( )
mmap	generic_file_mmap( )
open	generic_open_file( )
release	ext2_release_file( )
fsync	ext2_sync_file( )
…	…