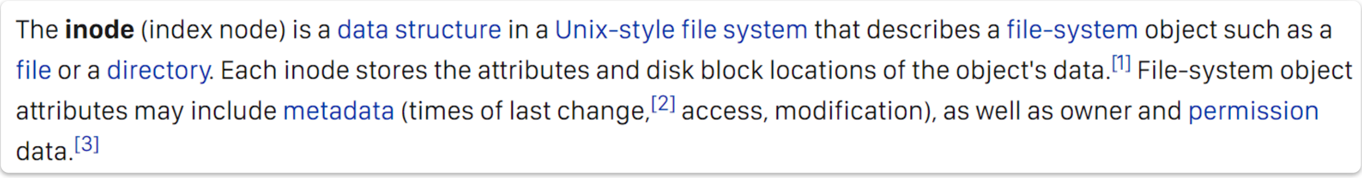
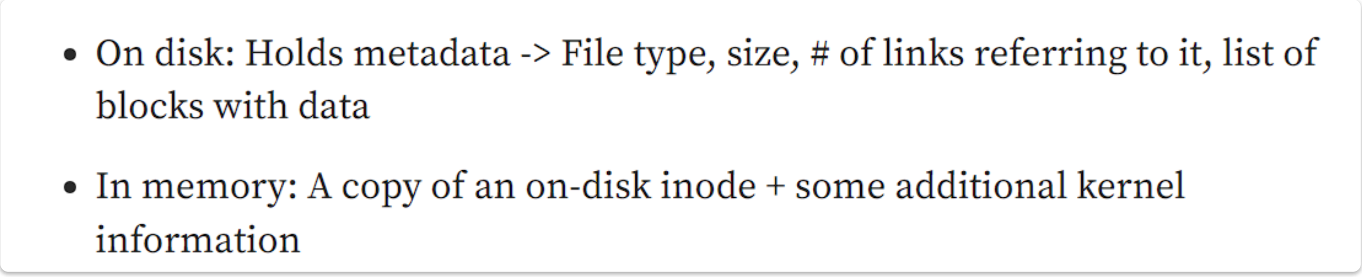
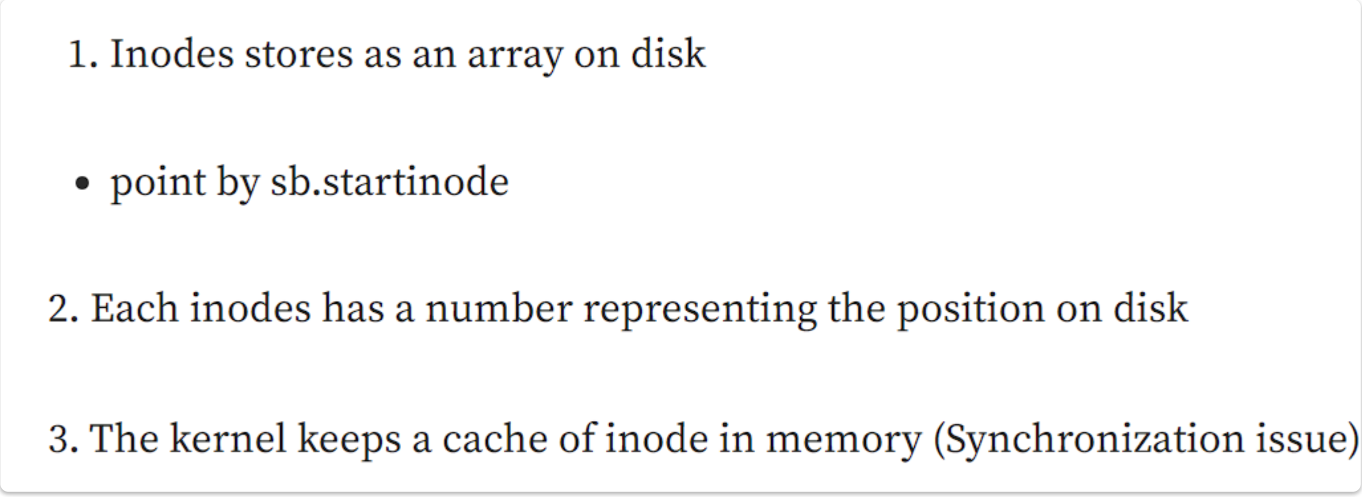
Inode(Index Node)



The **inode** (index node) is a data structure in a Unix-style file system that describes a file-system object such as a file or a directory. Each inode stores the attributes and disk block locations of the object's data.[1] File-system object attributes may include metadata (times of last change,[2] access, modification), as well as owner and permission data.[3]



- On disk: Holds metadata -> File type, size, # of links referring to it, list of blocks with data
- In memory: A copy of an on-disk inode + some additional kernel information



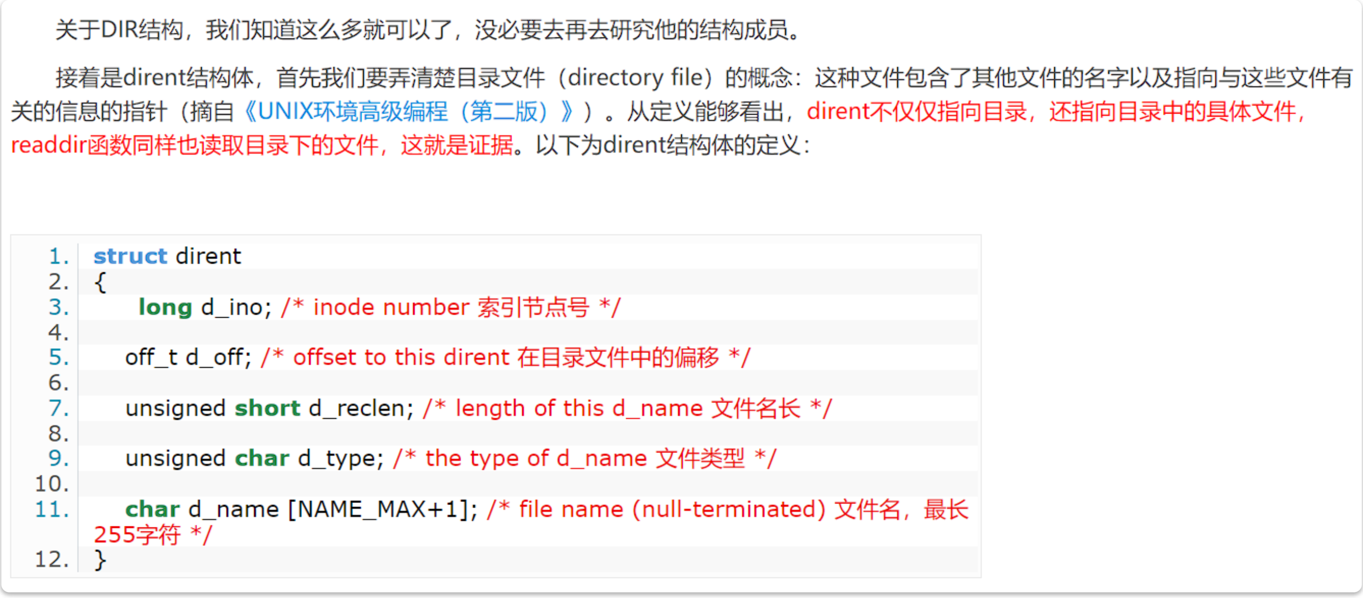
1. Inodes stores as an array on disk

- point by sb.startinode

2. Each inodes has a number representing the position on disk

3. The kernel keeps a cache of inode in memory (Synchronization issue)

dirent(directory entry)



关于DIR结构，我们知道这么多就可以了，没必要去再去研究他的结构成员。

接着是dirent结构体，首先我们要弄清楚目录文件（directory file）的概念：这种文件包含了其他文件的名字以及指向与这些文件有关的信息的指针（摘自《UNIX环境高级编程（第二版）》）。从定义能够看出，dirent不仅仅指向目录，还指向目录中的具体文件，readdir函数同样也读取目录下的文件，这就是证据。以下为dirent结构体的定义：

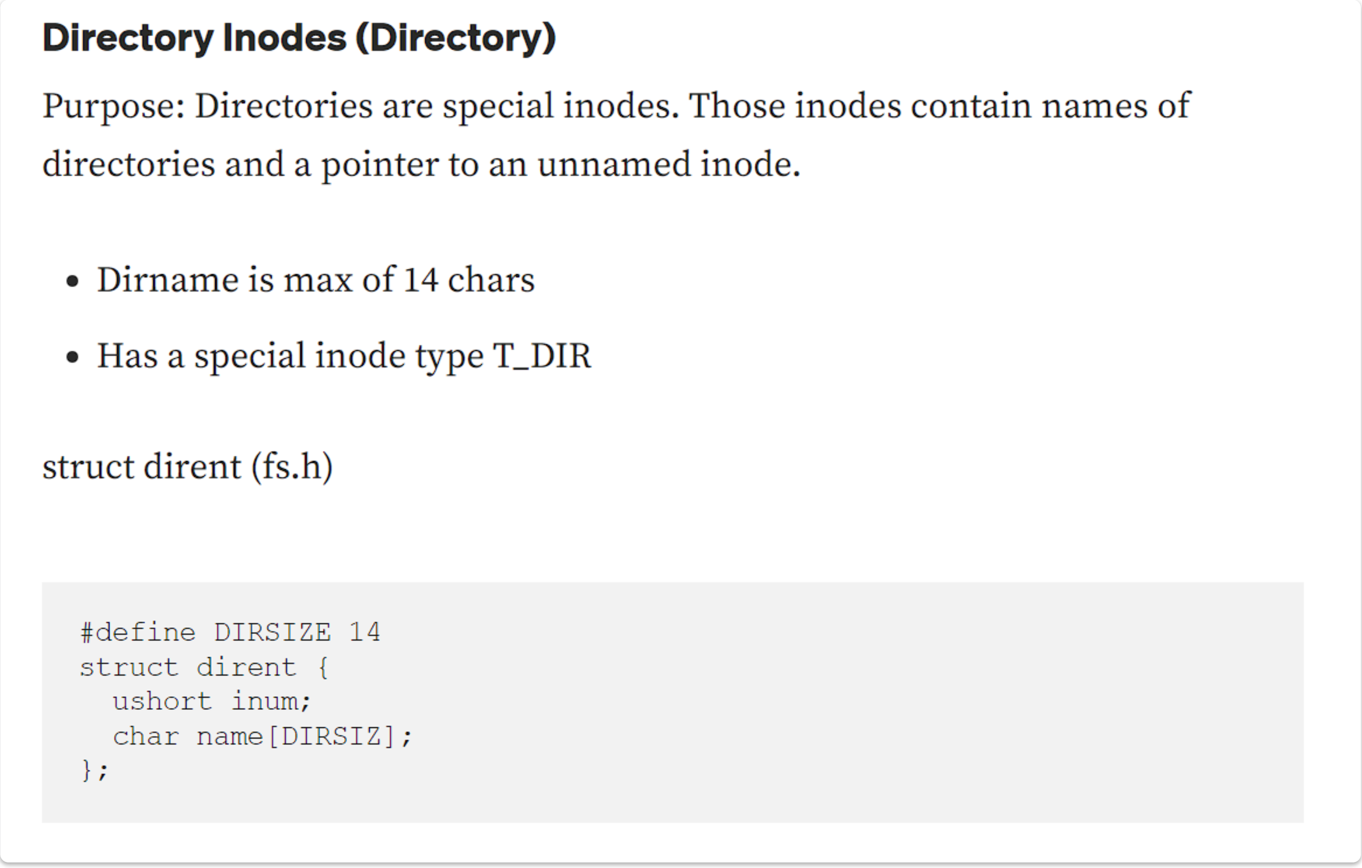
```
struct dirent
{
      long d_ino; /* inode number 索引节点号 */

    off_t d_off; /* offset to this dirent 在目录文件中的偏移 */

    unsigned short d_reclen; /* length of this d_name 文件名长 */

    unsigned char d_type; /* the type of d_name 文件类型 */

    char d_name [NAME_MAX+1]; /* file name (null-terminated) 文件名，最长255字符 */
}
```



Directory Inodes (Directory)

Purpose: Directories are special inodes. Those inodes contain names of directories and a pointer to an unnamed inode.

- Dirname is max of 14 chars
- Has a special inode type T_DIR

struct dirent (fs.h)

```
#define DIRSIZE 14
struct dirent {
  ushort inum;
  char name[DIRSIZ];
};
```