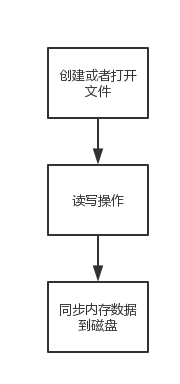
1、上位机程序在运行过程是一个有很多写文件数据库的过程。应用程序的读写流程比较简单。

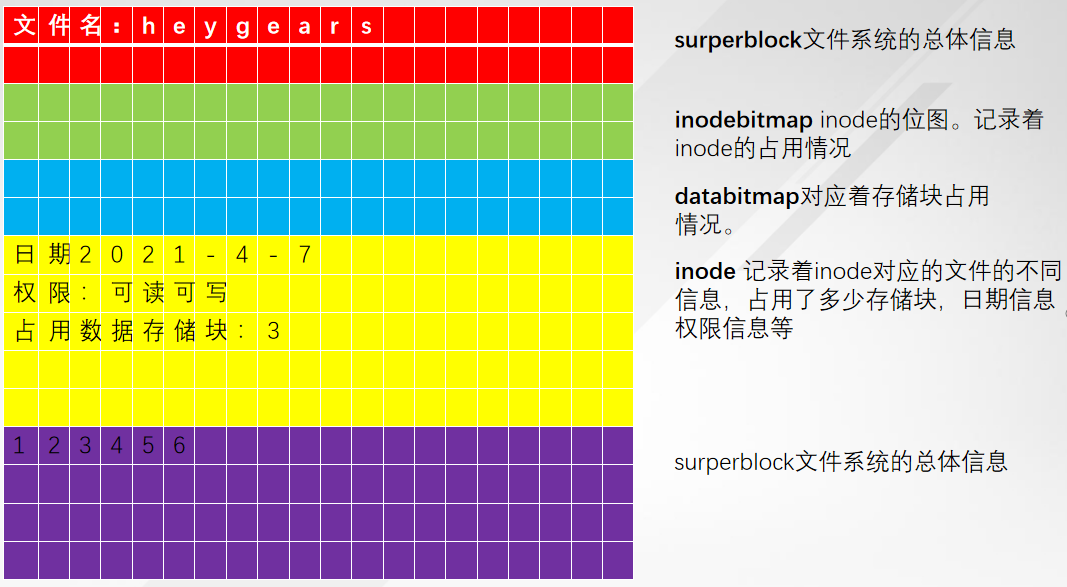


之前技术分享会有提到过这个文件读写的过程，直接拿出来说。

这个在系统底层去看上面的文件读写流程是一个很复杂的过程。如果将磁盘抽象成一个格子的话，文件的组成结构大致如下。

假设有个文件：

**名字heygears**，**修改时间2021年4月7日**，**权限可读可写可执行(wrx)**。则文件在磁盘的存储格式如下。（抽象的形式表现）



当你想改动文件的时候，对于linux来说要改动的地方很多。

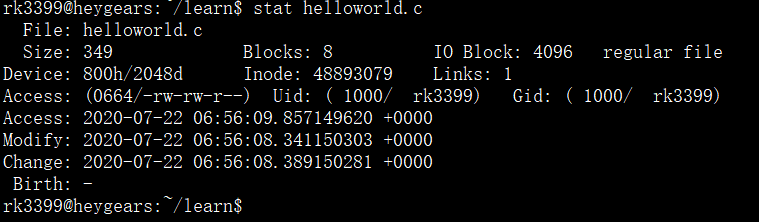
1、surperblock中的空闲内存块数量信息需要改动。

2、文件对应的inode信息占用的内存块多少以及日期要修改。

3、databitmap也需要更改。

4、文件写入首先是写进内存，但是内存什么时候写进硬盘，如果没有人为的干预,比如sync，那么这个时间这是系统决定的，这个时间不能太短，也不能太长，这个计算时间的过程是十分复杂的。

随便在系统查看一个文件



记录着文件名字helloworld.c，大小349个字节，占用的数据库空间数量8个，文件号48893079

权限0664 还有几个时间信息。

**A2D2.0-IMX6平台的侧键按键关键是硬件强制断电，如果在上诉写文件步骤中的任意一个环节断电了，则文件就会不一致，比如你写了文件名和文件时间，中途断电了数据没写进去，这时候这个文件就不一致了，文件就被损坏了**。

任何的软件技术都无法保证掉电不丢数据，我们只能保证数据的一致性（元数据+数据的一致性，或者元数据的一致性）、通常我们可以用fsck、和日志来提供一致性。

fsck是一个修复工具，一般当文件系统出问题后，利用fsck来修复该文件系统。A2D2.0文件系统损坏后用过这个工具修复过。但是这个时间非常慢。不太建议用在工程里面，而且很容易修复不成功。