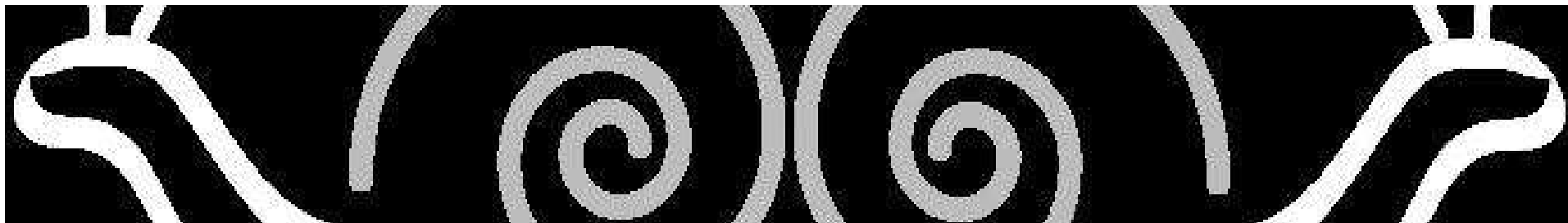


# 蜗窝科技

慢下来，享受技术。



[博客](#) [项目](#) [讨论区](#) [关于蜗窝](#) [联系我们](#) [支持与合作](#) [登录](#)

## 为什么会有文件系统(一)

作者：驴肉火烧 发布于：2017-1-17 20:56 分类：文件系统

Linux发展到现今，在fs目录下我们可以看到形形色色的文件系统，眼花缭乱的同时首先需要回答的问题是，为什么会有文件系统这个东西呢？我想如果能搞清楚这个问题，会帮助大家更好的理解文件系统，那么我就尝试着来模拟一次文件系统的演进过程，于是，我们来到了那一天，那天之前，人们还没有文件系统的概念。

**友情提示：**下面将在荒诞的场景下演进人类合理的诉求

神说，要有光，于是，光照大地

神说，要有风，于是，风动四方

神说，人类要记住神，于是，有了传说

神说，怕你们忘了，得记下来，于是，有了文字，信息被存储在石板上，竹片上，纸张上，硬盘里，flash中

### 站内搜索

### 功能

[留言板](#)

[评论列表](#)

[支持者列表](#)

### 最新评论

ctwillson

后续的是不是没了？

callme\_friend

@pixiandouban: visio

当信息能存在硬盘中的时候，人类如获至宝，如此大的存储量，我们能装下全世界图书馆的馆藏，于是，我们想先放一套盗墓笔记进去。

好嘞，于是，我一个字一个字的将精彩的内容顺序存储在硬盘中，终于，全套的盗墓笔记被存储在硬盘中了，还没来得及高兴，就傻眼了，我不想看秦岭神树，怎么办，这并难不倒我，略加思索，就能想到解决方案，因为是顺序存储的，从开始的地方一直读下去，当恰好跳过秦岭神树章节内容的时候，就做一个标记，记录已经跳过的字节数，下次再看的时候，就直接读到硬盘对应的位置即可，经过一番努力，我找到了并把这个字节数写在了一张纸条上，以便下次可以直接读取，避免一次次的遍历。

后来，我开始有点不耐烦了，因为这张纸条里面的内容越来越多，比如最后一章的位置，终极第一次出现的位置等等，有时我甚至记不住我需要寻找的标记是否在纸条中了，终于有一天，这张纸条丢了，我只能呵呵并且从心底认为，仅仅是顺序存储无法满足我的需求，我需要管理这些内容。

我想，最起码我需要能把全套的盗墓笔记分为8本书吧，只要根据书名，比如邛崃石影，我就立刻能找到对应的内容，我立刻想到了最简单的解决方案，仍然使用顺序存储，只不过在内容录入的时候，给每本书分100MB的存储空间，这样我如果想看第7本，那么直接从600MB偏移开始即可，那么一套盗墓笔记只需要800MB就可以存储，但是，我很快又有了一个更优的方案，在每本书的100MB可用空间内，再进行细分，给每章节进行划分，假设每本书有50章，那么每章节就是2MB空间，这样每章节按照2MB对齐，我要找第6本书的第30章节，就是 $(500 + 29 * 2)$ MB 偏移，我甚至都有点洋洋自得了，简单的设计一下就可以再也不用依赖那张小纸条(已遗失)了。

但是，很快我又遇到了新的挑战，因为这块硬盘不是我的，开始说好的800MB没有了，我被要求只能使用8MB来存储全套的盗墓笔记，原先的设计继续使用，每章只能分到20KB，这样有些内容多的章节会越界，而有些内容少的章节又不够饱满，那些没有被利用起来的空间此时显得是那么的珍贵，于是我开始了小心翼翼字斟句酌的重新设计。

callme\_friend

@hit201j: 太忙，没时间回复，  
需要的朋友可统一去以下链...

callme\_friend

@icecoder: 太忙，没时间回复，  
需要的朋友可统一去以下...

callme\_friend

@就爱吃泡芙: 太忙，没时间回  
复，需要的朋友可统一去以下链接  
下...

callme\_friend

@今雨轩: 太忙，没时间回复，需  
要的朋友可统一去以下链接下  
载: ...

## 文章分类

- Linux内核分析(11)
- 统一设备模型(15)
- 电源管理子系统(42)
- 中断子系统(15)
- 进程管理(19)
- 内核同步机制(18)
- GPIO子系统(5)
- 时间子系统(14)
- 通信类协议(7)
- 内存管理(27)
- 图形子系统(1)
- 文件系统(4)
- TTY子系统(6)
- u-boot分析(3)
- Linux应用技巧(13)
- 软件开发(6)
- 基础技术(13)
- 蓝牙(16)
- ARMv8A Arch(13)

看起来，顺序存储是最节约空间的，那么只有将小纸条(已遗失)的内容也存储在硬盘中了。于是，喝下一罐可乐后，我发觉将章节抽象成一个章节类是一个不错的注意，每个章节是该类的一个对象实例，类成员包括章节名称，章节起始位置，章节字数，每个对象都64字节对齐，这样400章的索引信息只需要25KB即可完成存储，我大大方方的将全部的章节类对象存储在8MB的前32KB区域，后面剩余的全部顺序存储内容，就这样，随着需求的不断增加，我的设计也渐渐开始有文件系统的影子了，尽管我并不知道，但是一切就这样发生了，是那么的自然。

原创文章，转发请注明出处。蜗窝科技，[www.wowotech.net](http://www.wowotech.net)

标签: 文件系统



« 进程切换分析(1): 基本框架 | eMMC 原理 2 : eMMC 简介 »

发表评论:

昵称

邮件地址 (选填)

个人主页 (选填)

显示(3)

USB(1)

基础学科(10)

技术漫谈(12)

项目专区(0)

X Project(28)

## 随机文章

Linux时间子系统之（十七）：  
ARM generic timer驱动代码分  
析  
ARMv8之Observability  
Linux内核同步机制之（六）：  
Seqlock  
Linux内核同步机制之（五）：  
Read/Write spin lock  
u-boot启动流程分析(2)\_板级  
(board)部分

## 文章存档

2018年10月(1)  
2018年8月(1)  
2018年6月(1)  
2018年5月(1)  
2018年4月(7)  
2018年2月(4)  
2018年1月(5)  
2017年12月(2)  
2017年11月(2)  
2017年10月(1)  
2017年9月(5)  
2017年8月(4)  
2017年7月(4)  
2017年6月(3)

bB5NA

发表评论

- 2017年5月(3)
- 2017年4月(1)
- 2017年3月(8)
- 2017年2月(6)
- 2017年1月(5)
- 2016年12月(6)
- 2016年11月(11)
- 2016年10月(9)
- 2016年9月(6)
- 2016年8月(9)
- 2016年7月(5)
- 2016年6月(8)
- 2016年5月(8)
- 2016年4月(7)
- 2016年3月(5)
- 2016年2月(5)
- 2016年1月(6)
- 2015年12月(6)
- 2015年11月(9)
- 2015年10月(9)
- 2015年9月(4)
- 2015年8月(3)
- 2015年7月(7)
- 2015年6月(3)
- 2015年5月(6)
- 2015年4月(9)
- 2015年3月(9)
- 2015年2月(6)
- 2015年1月(6)
- 2014年12月(17)
- 2014年11月(8)
- 2014年10月(9)
- 2014年9月(7)
- 2014年8月(12)
- 2014年7月(6)
- 2014年6月(6)

2014年5月(9)

2014年4月(9)

2014年3月(7)

2014年2月(3)

2014年1月(4)

