



链滴

关于 IO 的几种模式的讨论 (9.26 号更新)

作者: [tianxin](#)

原文链接: <https://ld246.com/article/1474770594232>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2 id="IO的技术讨论">IO 的技术讨论</h2>

<h2 id="9-26号的更新补充-增加新的看法-">9.26 号的更新补充（增加新的看法）</h2>

<p> 带着这个问题自己去看了早期 linux 内核的相关代码（因为设计的一个概念是一切皆文件，以从文件系统相关内容开始浏览），但是发现会从 虚拟文件系统 —— 文件系统 —— 缓冲模块 —— 相应的底层读写 这样的流程走下来，越来越偏向底层。从而产生了新的问题：</p>

<h3 id="最新问题--所以作为目前的大三学生来说-之前精力较多以Web开发为主--是否有必要花较多的精力来去学习linux相关内核-并去造一个小型的-OS-轮子呢-">最新问题： 所以作为目前的大三学来说（之前精力较多以 Web 开发为主），是否有必要花较多的精力来去学习 linux 相关内核 并去造个小型的 OS 轮子呢？</h3>

> PS：该问题已同时在知乎上提问，大家倘若有兴趣，可以在知乎上看一下其他人的观点：[[作一名学生，是否应该去花较多的精力看linux内核的源码呢？然后再重新造一个小型的轮子。]](https://www.zhihu.com/question/51003849)](https://www.zhihu.com/question/51003849/answer/12380388?group_id=763728785135652864#comment-169676764)

<p>

</p>

<hr>

<h2 id="如果根据用户空间与磁盘之间的访问方式的差异来划分IO-我想应该可以划分为三大类">如根据用户空间与磁盘之间的访问方式的差异来划分 IO，我想应该可以划分为三大类</h2>

<h3 id="标准IO">标准 IO</h3>

<p>标准 IO 的过程： 用户地址空间 —— 内核地址空间 —— 磁盘空间</p>

<h3 id="直接IO">直接 IO</h3>

<p>直接 IO 的过程： 用户地址空间 —— 磁盘空间</p>

<h3 id="内存映射">内存映射</h3>

<p>内存映射的过程： 内核地址空间 —— 磁盘空间（并将用户空间的一段内区域映射到内核空间）</p>

<blockquote>

<p>PS：倘若不清楚用户地址空间与内核地址空间二者区别，可以查看操作系统相关原理：用户态与内核态的区别。</p>

</blockquote>

<h2 id="讨论问题1--这几种工作模式的使用场景是什么呢-">讨论问题 1：这几种工作模式的使用场景是什么呢？</h2>

<p> 自己目前只是做过一些以 Servlet 为核心的 Java Web 开发项目，这其中负责和前端传输的角色一般是：Tomcat 或者是 Jetty。之前有读过部分 Tomcat 源码，不论是 NIO 还是 BIO（未看过 Apr 模块）都是符合 标准 IO 的工作方式：用户地址空间 —— 内核地址空间 —— 磁盘空间（只不过 NIO 的设计是采用了多路复用的思想）。所以由于自身眼界有限，并未接触过 直接 IO 和 内存映射这两种开发使用场景，请大家多多指教和讨论哈。</p>

<h2 id="讨论问题2---不同的IO模型与上述IO工作机制的联系是什么呢-">讨论问题 2：不同的 IO 型与上述 IO 工作机制的联系是什么呢？</h2>

<p>IO 模型的划分主要是分成以下几类：</p>

<p>阻塞 IO（BIO）（这里不再讨论非阻塞 IO）</p>

<p>多路复用 IO（NIO）</p>

<p>异步 IO（AIO）</p>

<p>这些不同的 IO 模型和上述的 IO 机制之间又有怎样的联系呢？</p>

<blockquote>

<p>个人目前理解：BIO 和 NIO 都是符合标准 IO 的工作机制。</p>

</blockquote>

<p>欢迎大家讨论哈 ~ ~ ~ </p>