<https://blog.csdn.net/xxxxxx91116/article/details/51233785>

**【高并发编程】再谈同步、异步、阻塞、非阻塞**

2016年04月24日 16:13:35 [xxxxxx91116](https://me.csdn.net/xxxxxx91116) 阅读数：10895 标签： [异步](http://so.csdn.net/so/search/s.do?q=%E5%BC%82%E6%AD%A5&t=blog)[同步](http://so.csdn.net/so/search/s.do?q=%E5%90%8C%E6%AD%A5&t=blog)[阻塞](http://so.csdn.net/so/search/s.do?q=%E9%98%BB%E5%A1%9E&t=blog)[非阻塞](http://so.csdn.net/so/search/s.do?q=%E9%9D%9E%E9%98%BB%E5%A1%9E&t=blog) 更多

个人分类： [操作系统开发篇](https://blog.csdn.net/xxxxxx91116/article/category/1282798)

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/xxxxxx91116/article/details/51233785

同步、异步、阻塞、非阻塞的概念一直是计算机学科中很重要的概念，而这种细微的差别常常被大家混淆，我自己在过一段时间后也需要复习。今天再次翻出这个概念，仍然觉得不够清晰，今天再次深入了解了这四大天王。

以前转过一篇博客：http://blog.csdn.net/xxxxxx91116/article/details/12083613

但是始终太抽象，每次都要重头看，其次，这篇文章仍然没有很清晰的表达四大天王的特色，因此这里给出自己的分析：

一、一个有问题，但有趣的例子

在网上看到一个有趣的例子，虽然我认为他的表述仍然是有问题的，但是有助于我们的理解：

|  |
| --- |
| 老张爱喝茶，废话不说，煮开水。 出场人物：老张，水壶两把（普通水壶，简称水壶；会响的水壶，简称响水壶）。 1 老张把水壶放到火上，立等水开。（同步阻塞） 老张觉得自己有点傻 2 老张把水壶放到火上，去客厅看电视，时不时去厨房看看水开没有。（同步非阻塞） 老张还是觉得自己有点傻，于是变高端了，买了把会响笛的那种水壶。水开之后，能大声发出嘀~~~~的噪音。 3 老张把响水壶放到火上，立等水开。（异步阻塞） 老张觉得这样傻等意义不大 4 老张把响水壶放到火上，去客厅看电视，水壶响之前不再去看它了，响了再去拿壶。（异步非阻塞） 老张觉得自己聪明了。   所谓同步异步，只是对于水壶而言。 普通水壶，同步；响水壶，异步。 虽然都能干活，但响水壶可以在自己完工之后，提示老张水开了。这是普通水壶所不能及的。 同步只能让调用者去轮询自己（情况2中），造成老张效率的低下。  所谓阻塞非阻塞，仅仅对于老张而言。 立等的老张，阻塞；看电视的老张，非阻塞。 情况1和情况3中老张就是阻塞的，媳妇喊他都不知道。虽然3中响水壶是异步的，可对于立等的老张没有太大的意义。所以一般异步是配合非阻塞使用的，这样才能发挥异步的效用。  作者：愚抄 链接：http://www.zhihu.com/question/19732473/answer/23434554 来源：知乎 著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。 |

二、上述例子的问题和理论分析

**上面的例子很形象吧？可以我认为他是有问题的，问题在哪里呢？**

先给出三张不同模式下的系统交互图：

A、同步阻塞模式

B、同步非阻塞模式

C、异步非阻塞模式

好的，看完这三个图，你应该在想，这是什么鬼，不想看了。再听我啰嗦2句，你就可以跳到下面看小张和狗狗的故事了。

从上面的图可以看出来，其实不管是任何模式，内核的处理方式都是一样的，都是

1. wait for data（等待数据）；

2. copy data from kernel to user（拷贝数据到用户层）。

而产生这三种模式的差别在于中间使用的api函数不同。

这和上面的小故事可是有一些差别的。上面的例子中，小张相当于我们的应用程序，而响水壶和普通水壶相当于内核。这就有问题了，其实我们的内核都是相同的，并不存在响水壶和普通水壶的差别，那么怎么样描述才正确呢？

**三、正确的例子（我很好玩！）**

小张喜欢喝咖啡，同时养了好多狗；

出场：

1. 小张：相当于我们的客户端进程

2. 小狗大黑：阻塞处理的IO函数

3. 小狗大黄：非阻塞处理的IO函数

4. 小狗大白、大红：异步处理的IO函数

同步阻塞：

小张派大黑去看咖啡煮好没，大黑等咖啡煮开了才回来；

同步非阻塞：

小张派大黄去看咖啡煮好没，大黄看了一眼就回来了，过了一会，再大黄再去看看咖啡煮好没；

异步非阻塞：

小张派大白和大红去看咖啡煮好没，大白和大红到了厨房后，大白就回来告诉小张，大红已经到厨房啦；过了一会咖啡煮好了，大红回到客厅告诉小张

异步阻塞：（这个太傻了，目前还没遇到）

小张派大白和大红去看咖啡煮好没，大白和大红到了厨房后，一起在那等着；过了一会咖啡煮好了，大红大白一起回到客厅告诉小张

那么采用何种方式，要看小张有没有养这个类型的狗（系统有没有相关API），在有的情况下，就看小张个人的需求了（业务需求）