**按照银行家算法的思想，当进程请求资源时，系统将按如下原则分配系统资源：**

(1) 当一个进程对资源的最大需求量不超过系统中的资源数时可以接纳该进程。

(2) 进程可以分期请求资源，当请求的总数不能超过最大需求量。

(3) 当系统现有的资源不能满足进程尚需资源数时，对进程的请求可以推迟分配，但总能使进程在有限的时间里得到资源。

(4) 当系统现有的资源能满足进程尚需资源数时，必须测试系统现存的资源能否满足该进程尚需的最大资源数，若能满足则按当前的申请量分配资源，否则也要推迟分配。

[浅谈HTTP中Get与Post的区别](http://www.cnblogs.com/hyddd/archive/2009/03/31/1426026.html)

HTTP中的GET，POST，PUT，DELETE就对应着对这个资源的查，改，增，删4个操作

　1.根据HTTP规范，GET用于信息获取，而且应该是安全的和幂等的。

　　(1).所谓安全的意味着该操作用于获取信息而非修改信息。换句话说，GET 请求一般不应产生副作用。就是说，它仅仅是获取资源信息，就像数据库查询一样，不会修改，增加数据，不会影响资源的状态。

\* 注意：这里安全的含义仅仅是指是非修改信息。

但在实际应用中，以上2条规定并没有这么严格。比如，新闻站点的头版不断更新。虽然第二次请求会返回不同的一批新闻，该操作仍然被认为是安全的和幂等的，因为它总是返回当前的新闻。从根本上说，如果目标是当用户打开一个链接时，他可以确信从自身的角度来看没有改变资源即可。

　　(2).幂等的意味着对同一URL的多个请求应该返回同样的结果。

　2.根据HTTP规范，POST表示可能修改变服务器上的资源的请求。

继续引用上面的例子：还是新闻以网站为例，读者对新闻发表自己的评论应该通过POST实现，因为在评论提交后站点的资源已经不同了，或者说资源被修改了。

表面现像上面看看GET和POST的区别：

**1**.GET请求的数据会附在URL之后（就是把数据放置在HTTP协议头中），以?分割URL和传输数据，参数之间以&相连，如：login.action?name=hyddd&password=idontknow&verify=%E4%BD%A0%E5%A5%BD。如果数据是英文字母/数字，原样发送，如果是空格，转换为+，如果是中文/其他字符，则直接把字符串用BASE64加密，得出如：%E4%BD%A0%E5%A5%BD，其中％XX中的XX为该符号以16进制表示的ASCII。

　　POST把提交的数据则放置在是HTTP包的包体中。

**2**."GET方式提交的数据最多只能是1024字节(这个限制是特定的浏览器及服务器对它的限制, URL不存在参数上限的问题)，理论上POST没有限制，可传较大量的数据，IIS4中最大为80KB，IIS5中为100KB(限制作用的是服务器的处理程序)"

**3**.POST的安全性要比GET的安全性高。注意：这里所说的安全性和上面GET提到的“安全”不是同个概念。上面“安全”的含义仅仅是不作数据修改，而这里安全的含义是真正的Security的含义，比如：通过GET提交数据，用户名和密码将明文出现在URL上，因为(1)登录页面有可能被浏览器缓存，(2)其他人查看浏览器的历史纪录，那么别人就可以拿到你的账号和密码了，除此之外，使用GET提交数据还可能会造成Cross-site request forgery攻击。在FORM（表单）中，Method默认为"GET"，实质上，GET和POST只是发送机制不同，并不是一个取一个发

* GET请求能够被缓存
* GET请求会保存在浏览器的浏览记录中
* 以GET请求的URL能够保存为浏览器书签
* GET请求有长度限制
* GET请求主要用以获取数据
* POST请求不能被缓存下来
* POST请求不会保存在浏览器浏览记录中
* 以POST请求的URL无法保存为浏览器书签
* POST请求没有长度限制

DNS解析过程：  
1.现在我有一台计算机，通过ISP接入了互联网，那么ISP就会给我分配一个DNS服务器，**这个DNS服务器不是权威服务器**，而是相当于一个代理的dns解析服务器，他会帮你迭代权威服务器返回的应答，然后把最终查到IP返回给你。

2.现在的我计算机要向这台ISPDNS发起请求查询www.baidu.com这个域名了，(经网友提醒：这里其实准确来说不是ISPDNS，而应该是用户自己电脑网络设置里的DNS，并不一定是ISPDNS。比如也有可能你手工设置了8.8.8.8)

3.ISPDNS拿到请求后，先检查一下自己的缓存中有没有这个地址，有的话就直接返回。这个时候拿到的ip地址，会被标记为**非权威服务器的应答**。

4.如果缓存中没有的话，ISPDNS会从**配置文件**里面读取13个根域名服务器的地址（这些地址是不变的，直接在BIND的配置文件中），

5.然后像其中一台发起请求。

6.根服务器拿到这个请求后，知道他是com.这个顶级域名下的，所以就会返回com域中的NS记录，一般来说是13台主机名和IP。

7.然后ISPDNS向其中一台再次发起请求，com域的服务器发现你这请求是baidu.com这个域的，我一查发现了这个域的NS，那我就返回给你，你再去查。

（目前百度有4台baidu.com的顶级域名服务器）。

8.ISPDNS不厌其烦的再次向baidu.com这个域的权威服务器发起请求，baidu.com收到之后，查了下有www的这台主机，就把这个IP返回给你了，

9.然后ISPDNS拿到了之后，将其返回给了客户端，并且把这个保存在高速缓存中。