主要的[**测试**](javascript:;)思路是借鉴以上两篇文章.  
需要写[**java**](javascript:;)代码，打成jar包，放在Jmeter的lib相关目录.  
虽然有借鉴先辈们的经验，还是有自己遇到的问题，以下就简单罗列下：  
1.查看自己Jmeter的版本，查看JmeterHome/lib/ext目录下是否有  
Jmeter\_java.jar  
Jmeter\_core.jar  
这两个包的引入在上面第二篇文章中也有提到，如果Jmeter版本不对可能就需要下载新的版本使用。

2.查看JmeterHome/lib目录是否有  
jorphan.jar  
logkit-2.0.jar  
avalon-framework-4.1.4.jar  
这三个包是在代码编完后运行抛错时才发现的，在Jmeter的lib下找，然后导入到相应工程。

3.先在eclipse上测试下zk的连接、creat、delete、set等方法是否能正常工作。  
导出为jar包。export--Jar File--Next  
勾选export generated class files and resources  
export java source files and resources  
Next  
finish

4.将Jar存放在JmeterHome/lib/ext目录下，重启Jmeter  
创建线程组---创建Java请求--配置参数---运行

测试ZK读写效率的思路：  
如第一篇文章，我首先测试节点的创建和删除的压力，  
create node 、delete node。不断增大线程数，并记录响应时间、TPS（每秒完成的请求数），以观察处理情况。  
我分别做了1000、5000、10000的测试：90%用户响应时间都在20--24ms，TPS 10/s

第二个测试加入了watcher  
创建节点，并对节点进行监控，修改节点后收到节点变化的通知，然后读取变化的数据