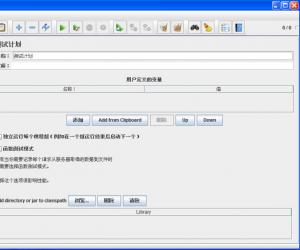
1. jmeter环境准备及运行  
jmeter是Apache 组织的开放源代码项目，是一个100%纯**[java](javascript:;" \t "_self)**桌面应用，主要用于压力**[测试](javascript:;" \t "_self)**和性能测试  
jmeter下载地址：http://jmeter.apache.org/download\_jmeter.cgi  
安装需求：  
1.1 java版本：在JVM1.5或者更高版本上（java环境配置此处不多?嗦）  
1.2 [**操作系统**](javascript:;)：可以在任何一个部署了java的系统上运行  
1.3 jmeter下载完成后，放置某个目录下，解压，只需运行bin目录下：jmeter.bat（也可以建立桌面快捷方式）就可以运行jmeter，运行界面如下



2. jmeter学习  
2.1 基本概念  
1. 测试计划：是使用jmeter进行测试的起点，它是其它jmeter测试元件的容器  
2. 线程组：代表一定数量的并发用户，可以用来模拟并发用户的发送请求。实际的请求内容再sampler中定义，它被线程组包含，可以在测试计划-->添加-->线程组来建立，在线程组面板里有几个输入栏：线程数、Ramp-Up Period(in seconds)、循环次数，其中Ramp-Up Period(in seconds)表示在这时间内创建完所有的线程。如有8个线程，Ramp-Up = 200秒，那么线程的启动时间间隔为200/8=25秒，这样的好处是：一开始不会对服务器有太大的负载。线程组是为模拟并发负载而设计。  
3. 取样器（Sampler）：模拟各种请求。所有实际的测试任务都由取样器承担，存在很多种请求。如：HTTP请求  
4.监听器：负责收集测试结果，同时也被告知了结果显示的方式。功能是对取样器的请求结果显示、统计一些数据（吞吐量、KB/S……）等。  
5.断言：用于来判断请求响应的结果是否如用户所期望，是否正确。它可以用来隔离问题域，即在确保功能正确的前提下执行压力测试。  
6.定时器：负责定义请求（线程）之间的延迟间隔，模拟对服务器的连续请求。  
7. 逻辑控制器：允许自定义JMeter发送请求的行为逻辑，它与Sampler结合使用可以模拟复杂的请求序列。  
8.配置元件维护Sampler需要的配置信息，并根据实际的需要会修改请求的内容。  
9. 前置处理器和后置处理器负责在生成请求之前和之后完成工作。前置处理器常常用来修改请求的设置，后置处理器则常常用来处理响应的数据。

3. 关于Aggregate Report （聚合报告）解析  
Aggregate Report是JMeter常用的一个Listener，中文被翻译为“聚合报告”。  
如果是做Web应用的性能测试，例如只有一个登录的请求，那么在Aggregate Report中，会显示一行数据，共有10个字段，含义分别如下。  
Label：每个JMeter的element（例如HTTP Request）都有一个Name属性，这里显示的就是Name属性的值  
#Samples：表示你这次测试中一共发出了多少个请求，如果模拟10个用户，每个用户迭代10次，那么这里显示100  
Average：平均响应时间——默认情况下是单个Request的平均响应时间，当使用了Transaction Controller时，也可以以Transaction为单位显示平均响应时间  
Median：中位数，也就是50％ 用户的响应时间  
90% Line：90％ 用户的响应时间  
Note：关于50％ 和90％ 并发用户数的含义，  
Min：最小响应时间  
Max：最大响应时间  
Error%：本次测试中出现错误的请求的数量/请求的总数  
Throughput：吞吐量——默认情况下表示每秒完成的请求数（Request per Second），当使用了Transaction Controller时，也可以表示类似LoadRunner的Transaction per Second数  
KB/Sec：每秒从服务器端接收到的数据量，相当于LoadRunner中的Throughput/Sec