# Jmeter实例

# 基础

1. 1.Jmeter可以对（文件，接口，java对象，数据库，ftp服务器等）进行不同压力条件下的性能测试，也提供图形化界面。
2. 2.Jmeter 和 loadrunner 的区别

都支持：a.脚本录制

B.参数化

C.集合点 检查点

D.关联 多协议

E.多线程

F.生成报告导出

Jmeter体积小，便捷，开源。

Loadrunner 体积大，商业，成本高。

1. 工具组成部分

资源生产器（生成代码）

用户运行器（脚本运行）

报表生成器。

负载发生器。

1. 八类可以执行的原件

配置元件, 定时器, 前置处理器, 后置处理器, 断言, 监听器, 取样器, 逻辑控制器

1. 取样器：性能测试中向服务器发送请求，记录响应信息。
2. badboy 脚本录制器

实例：

测试百度平均响应时间（负载：30QPS 每秒查询率），测试服务器性能。

**实例**

**测试搜狗**

【测试百度】

1.打开badboy ，输入www.baidu.com

2.点下红色录制按钮。做一下操作。按下停止键。

3.可以在step里面播放刚才录制的脚本动作。

4. File》export to Jmeter

5.打开jmeter, 打开刚才录制的脚本\*.jmx

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

badboy 检查点，参数化

【测试sogou】

为什么不测试百度呢？因为百度的脚本复杂（每输入一个字符都会刷新），步骤很多很烦，断言也可能会出错。目前断言只用英文测试，中文可能会出错。

==================

[单个参数]检查点：

1.准备www.sogou.com,按录制，点下 ->

2.输入“nice”,点击搜索，选择“nice”》tools》arsert for slection(此时左边会出现一个问号)

3.输入框输入‘job’,点击搜索，同上加入断言。

4.点击搜索

5.点击新闻

6.点击“停止”键

右键step1->play.(观察check for text的地方有没有打钩)

==================

参数化+检查点：

1.Variables选项卡》add Variable 》name：sw 加入 cctv1， cctv2 ，2个

2.准备www.sogou.com,按录制，点下 ->

3.输入“test”，右键“/web”(找到query=test)，右键query点击properties，设置value中值${sw}变量。

4.右键step1中properties，设置for each value中的sw循环便利。

5.点击“停止”键

6.在最后添加检查点，在检查点里面的properties中expression，改成${sw}

右键step1->play.(观察有没有循环2次 分别cctv1和cctv2， 检查点是不是正确)

**测试百度QPS【基础，断言，参数化，集合点】**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

设置

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.添加线程组

线程数：虚拟用户数（比如20）

准备时长：全部虚拟用户启动所需时间。(时长=10，线程数20，则每秒启动2个线程)

循环次数：每个线程发送的请求次数。(次数=5，则总请求=20\*5=100)

2.添加HTTP请求：

添加》sampler（取样器）》http请求

名称：visit\_baidu (我这里设置中文会死机)

IP：www.baidu.com

3.设置QPS限制

添加常数吞吐量定时器

添加》定时器》constant timer （每分钟1800= 每秒30）

设置All active threads（分配在每个活跃线程中）

4.添加监听器

获得测试结果（这里我们关系的是请求响应时间）

在取样器》添加》监听器》聚合报告

在取样器》添加》监听器》察看结果树

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

启动，查看结果

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5.点击run， 保存线程组 visit\_baidu (不要有空格)

6.查看结果树

响应数据（网址返回的html，百度中文会有乱码，加密？？ 如果用www.51zxw.net显示正常）

7.聚合报告

sample：一共发送了多少个请求。

average：平均响应时间。

median:中位数，50%用户响应时间

90% line:90%用户响应时间

min max：最小 最大响应时间

error：错误量

throughput：吞吐量（每秒完成的请求数）

kb/sec:每秒从服务器接受的数据量

（报告可以写入文件。baidu的速度明显比51test快， 扫把按钮是清楚结果）

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

断言

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8.添加断言

在取样器》添加》断言》响应断言（设置一个检查点，加入“百度一下”，复制进去 中文直接输入会死机）（可以禁用和启用）

在取样器》添加》监听器》断言结果

在取样器》添加》断言》断言持续时间（单位毫秒，设置这个时间如果小于实际时间，就会failed）

（断言禁止）

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

参数化

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

9.添加用户参数

在取样器》添加》前置处理器》用户参数（添加 名称：wd，用户: 你好）

在取样器》http请求》路径里面添加 ： /s (这个可以看下百度搜索时的协议状态https://www.baidu.com/s?wd=你好)

在取样器》parameters》添加 名称：wd，值： ${wd}

(查看结果树》请求 GET http://www.baidu.com/s?wd=百度，查看响应数据)

10.使用CSV配置

禁用用户参数。删除取样器para里的参数

创建D:\table.txt (第一个参数wd=搜索内容，第二个参数rn=每页显示条数)

你好,1

百度,2

在取样器》添加》配置元件》CVS Data set config

filename：D:\table.txt

variable:wd,rn delimiter: ,(用逗号区分)

取样器》parameters》添加 名称：wd，值： ${wd}，名称：rn，值： ${wd}

线程》循环次数：2（2行）

（察看结果树验证：GET http://www.baidu.com/s?wd=你好&rn=1 ， GET http://www.baidu.com/s?wd=百度&rn=2）

11.随机参数化

选项》函数助手》选个random（输入最小1，最大5，点下生成，会出来 ${\_\_Random(1,5,)}）

${\_\_Random(1,5,)} 替换（刚才，取样器》parameters》添加值的） ${wd}

（察看结果树验证：GET http://www.baidu.com/s?wd=你好&rn=随机数）

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

集合点

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

理解为“多用户并发测试”（类似秒杀，大量户集中在同一时间访问），但真正的并发是不存在的，为了更真实的实现并发这感念，我们可以在需要压力的地方设置集合点，每到输入用户名和密码登录时，所有的虚拟用户都相互之间等一等，然后，一起访问。

12.添加集合器

在取样器》添加》定时器》sync timer （完成用要拖到取样器前面）（user=10个人，timeout=1000毫秒，到了10个人延迟1秒后开始）

**测试中国天气网【关联 正则表达式】**

用户登录后，session信息都不同，有些操作要使用session，经常遇到的场景，第二个请求提交的参数要从第一个请求的返回数据中获取。

需要先通过城市名字获取城市代码（接口http://toy1.weather.com.cn/search?cityname=“上海”）

通过城市代码获取城市天气信息（接口http://www.weather.com.cn/data/cityinfo/101020100.html 后面的数字为城市代码）

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

步骤

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.在测试计划下添加一个变量：名称：city ， 值： 上海

2.添加第一个HTTP Sampler， 命名为GetCityCode【下面2-7添加的内容都要在这个取样器下层中】

3.添加HTTP信息头管理器（配置元件里），再填写请求信息。 （Referer，http://www.weather.com.cn）

4.在http请求中设置：IP：toy1.weather.com.cn 路径 search?cityname=${city} （此时运行后可以在响应数据中看到上海的代码）

5.添加一个后置处理器（正则表达式提取器，改名为‘提取城市代码’），利用正则提取器将获取到的城市代码取出。

引用名称：citycode （下一个请求引用的参数名称）

(\d{9})~

正则表达式："ref":"(.+?)~shanghai (\d{9})~ 2个都可以，注意不要复制空格进去了。（实际返回的数据：【"ref":"101020100~shanghai】）

()：括起来的部分就是要提取的。

.：匹配任何字符串。

+：一次或多次。

?：不要太贪婪，在找到第一个匹配项后停止。

模板：$1$ (取第一个匹配到的值)

匹配数字：1 （表示取全部的值）

6.添加BeanShell断言 (if里面是数字 打印的时候${citycode}要加""变成字符 否则出错。语法很严谨，且无法检查，要仔细。 正则表达式获取返回代码有特殊字符~{}也会导致出错，要再学习。长\*\*\*为了log突出)

if(${citycode}==101020100){

log.info("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* right good:"+"${citycode}");

}

else{

log.info("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* bad"+"${citycode}");

Failure=true;

FailureMessage="获取失败，实际获取："+"${citycode}";

}

7.添加断言结果

（运行，打开选项中log，观察结果）

8.添加一个新的HTTP Sampler， 命名为GetWeatherInfo

在http请求中设置：IP：www.weather.com.cn 路径 /data/cityinfo/${citycode}.html （此时citycode传了下去）

9.添加察看树, bashshell断言，添加断言结果

引用名称：result (\d{9})~

正则表达式："city":"(.+?)" 实际返回的数据：【{"weatherinfo":{"city":"上海","cityid":"101020100","temp1":"6℃","temp2":"16℃","weather":"多云","img1":"n1.gif","img2":"d1.gif","ptime":"18:00"}}】

模板：$1$ (取第一个匹配到的值)

匹配数字：1 （表示取全部的值）

多增加一个正则：temp ， "temp1":"(.+?)"

if("${result}"=="上海"){

log.info("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 测试成功:"+"${result}"+" 温度："+"${temp}");

}

else{

log.info("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 测试失败:"+"${result}");

Failure=true;

FailureMessage="获取失败，实际获取："+"${result}";

}

log 输出 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 测试成功:上海 温度：6℃

**Mysql**

1.在测试计划主页面中加入驱动：add directory 浏览 选择jdbc

2.线程组》添加》配置元件》Jdbc connect

variable name ： mysql

database URL: jdbc:mysql://127.0.0.1:3306/samp\_db?serverTimezone=UTC (注意： samp\_db是数据库名字 后面的UTC不加的话会出现timezone错误)

jdbc driver class: com.mysql.jdbc.Driver

user: root

password: abc

3.添加jdbc取样器

线程组》添加》sampler》JDBC request

variable name ： mysql

Query：SELECT \* FROM EMPLOYEE; （EMPLOYEE是表名）

4.添加查看树 （run ，响应数据是数据表返回）

1. 添加响应断言（比如加入字符user\_1），断言结果（就是看有没有这个字符）

**Jmeter-Jenkins-ant自动化测试平台**

持续集成：

每天可能发生多次集成，通过自动化平台快速验证，尽快发现集成错误。

项目构建

对多个Jmeter脚本批量运行。

Ant

对Jmeter脚本批量编译

Jenkins

是一个开源的持续集成(CI)工具

安装配置：

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Ant

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. copy apache-ant-1.9.11 到 c:\

2. 设置环境变量

ANT\_HOME C:\apache-ant-1.9.11

path C:\apache-ant-1.9.11\bin;

classpath C:\apache-ant-1.9.11\lib;(这个要新建)

3. 将“C:\Program Files\Java\jdk\lib”目录下的tools.jar文件拷贝到“C:\Program Files\Java\jre\lib”下

(验证 ant verison ， 返回Buildfile: build.xml does not exist!Build failed 说明安装成功 failed可以先不管)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

jenkins

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

4.安装jenkins（安装完如果8080端口被别的应用占用，则需要修改xml的配置文件，没被占用就不用管，复制到外面改好在复制进去）

5.浏览器 http://localhost:8080/jenkins （复制密码 在C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword里）

输出 404bug （据说是bug 不用管）

6.浏览器 http://localhost:8080 （选择安装推荐插件，注册的时候选择右下角“使用admin”这样就不用输入用户名等了）

修改admin密码，在里面右上角用户admin点击》设置》密码 输入abc

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

配置jmeter

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

7. 录制2个jmeter脚本，在jmeter目录下新建一个文件夹loadtest，把2个脚本放进去

8. 复制 C:\apache-jmeter-2.13\extras\ant-jmeter-1.1.1.jar 到 C:\apache-ant-1.9.11\lib 里 (这步是关键)

9. 构建 build\_jmeter.xml （已经写好，可以用pycharm打开, 复制到C:\apache-jmeter-2.13\loadtest，中）

里面的 \*.jmx 就是执行所有jmx文件

10. cd C:\apache-jmeter-2.13\loadtest , ant -buildfile build\_jmeter.xml (ant 编译运行后 返回BUILD SUCCESSFUL ，会出现jtl html png等报告文件)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

部署到持续集成平台jmeter

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

11.浏览器 http://localhost:8080 admin abc， 新建任务 loadtest\_jenkins（自由风格项目），找到构建

12.下拉构建 invoke ant》高级》build file》C:\apache-jmeter-2.13\loadtest\build\_jmeter.xml 最下面保存

13.回到jenkins主页，点下loadtest\_jenkins右边的小三角， 选择立即构建。

左下角有执行状态，此时jenkins会通知ant去编译运行配置文件build.xml

刷新可以看到圆圈为success

点下“上次成功”可以看到控制台输出（和ant在命令行里面输出一样）

# 其他

1.分布式测试：Jmeter本身的短板，造成一台电脑模拟上千个用户会内存溢出， 所以可以多台电脑 Agent ，contrller模式测试。

2.java测试：先缓一缓，后面学java的时候做。

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

函数

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1.文件读取参数CSVRead()函数： 值里面写 ${\_\_CSVRead(D:\jlogins.txt,1)} 文件里第一个， ${\_\_CSVRead(D:\jlogins.txt,${\_\_Random(1,5,))})} 随机

2.StringFromFile ${\_StringFromFile(D:\table.txt,,,1)}

3.\_Counter (+1 计数功能)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

逻辑控制器

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

1. foreach controller : 读取多个自定义变量值

2. if controller：在条件里面设置。（比如线程数》2

3. interleave controller：交替控制器：依次访问各个节点（比如3次循环，把百度，我爱自学，\*\* 各执行以下）

4. Loop controller：直接循环运行 节点。

5. Once only controller： 只运行1次。比如登录。哪怕线程那里设置运行多次

6. random controller.

7. while controller.