+ 事务控制 + 聚合报告

+ jmeter中，默认一个取样器，发起一次请求就一个事务

+ 多个取样器要合并为一个事务，添加事务控制器

+ 聚合报告：

+ 1、没有网络瓶颈 吞吐量 tps

+ 2、并发用户数不变 聚合报告中的数据，是一个平均数

+ 每一行，代表一种事务

**+ Critical Section Controller 临界控制器 ----比较特殊，用的少**

+ 严格控制请求顺序

+ 锁名称： 默认是一个固定锁名称

+ 相当于把性能测试中的\*\*并行\*\*执行，强制转换为 \*\*串行\*\*

+ 如何让锁变成动态锁，生成多把锁？

+ 锁名称变成一个动态名称

+ global\_lock\_${\\_\\_threadNum}

${\\_\\_threadNum} 此函数为获取线程号这就导致运行时每次线程号不一致,由串行发送请求变更为并行发送请求

**+ Once Only Controller 仅一次控制器**

+ 仅一次控制器： 意思是，\*\*一个线程用户只执行一次\*\*

+ 理解？

+ 并发用户是10，不管你设置循环多少次，其下挂载的取样器，每个都只会执行10次

+ 并发用户是10，不管你运行多长时间，其下挂载的取样器，每个都只会执行10次

**+ 吞吐量控制器**

+ 总的吞吐量

+ 控制百分比， 使用多个吞吐量控制器，使他们的之和100

+ 不推荐大家用吞吐量控制器

+ 总请求量 = 并发用户数 \\* 时间 \\* 频率

**+ 随机控制器：**

+ 随机挑选其下挂载的取样器中1个执行

**+ 随机顺序控制器：**

+ 其下的所有取样器都会被打乱顺序执行

**+ 性能测试脚本开发：**

+ 1、写测试脚本

+ 2、可能需要添加 逻辑控制器（事务控制器）

+ 如果只有一个取样器，就不需要添加事务控制

+ 事务控制器，多个取样器合并时，才用

+ 3、性能场景设计

+ 普通

+ 负载

+ 压力

+ 混合

+ 面向目标

+ 波浪.....

+ 其他协议脚本开发

+ http协议

+ soap、jdbc、websocket、mq、dubbo

+ 现在的项目，一般，不会局限于一种协议

soap = http + xml

soap协议脚本本质就是 http协议加上xml数据格式

注意:

```xml

<?xml version="1.0" encoding="utf-8"?>

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">

<soap:Body>

<getSupportCityDataset xmlns="http://WebXml.com.cn/">

<theRegionCode>31123</theRegionCode>

</getSupportCityDataset>

</soap:Body>

</soap:Envelope>

```

```json

{"theRegionCode":"31123"}

```

soap 协议接口

混合场景： 不同并发用户数，使用不同接口发起请求

​ A: 30 -----吞吐量控制

​ B: 20

​ C: 10

