Performance Test
Simple Guide

Base on Jmeter

Author: Wangoo
Date: 2018-01-11

Jmeter?

- 100% java
- Master/Slave
- Open Source
- Request/Response

Jmeter Basic

- Test Plan：Test Project
- Thread Group： Thread count、Loop Count
- Config Element： CSV、User Define、Http Default、Http Header
- Timer: Constant 、Random
- Pre/Post processor: BeanShell Processor
- Listener：Aggregate Report、View Result、TPS
- Logic Controller：If、while、Loop、Random
- Sampler： HTTP、TCP、WebSocket、Java

Aggregate Report

Label ： 各个模拟测试的名称

#Samples ： 各个测试的样本总数

Average ： 每个请求的平均响应时间

Median ： 中值，即50%请求的平均响应时间

90%Line ： 90%请求的响应时间

Min ： 最小响应时间 ， Max ： 最大的响应时间

Error% ： 错误响应的概率。即无法响应的概率。

ThroughPut ： 吞吐量 -- 默认情况下表示每秒完成的请求数（Request per Second）。

KB/Sec ： 每秒从服务器端接收到的数据量。

Function&Variable

引用变量：${VARIABLE}

设置变量： vars.put(“name","value")

函数调用： ${__functionName(var1,var2,var3)}

属性设置：
${__setProperty(game_id, ${game_id})};
${__setProperty(waiting_all_charge, true)};

引用属性：
${__property(game_id)}

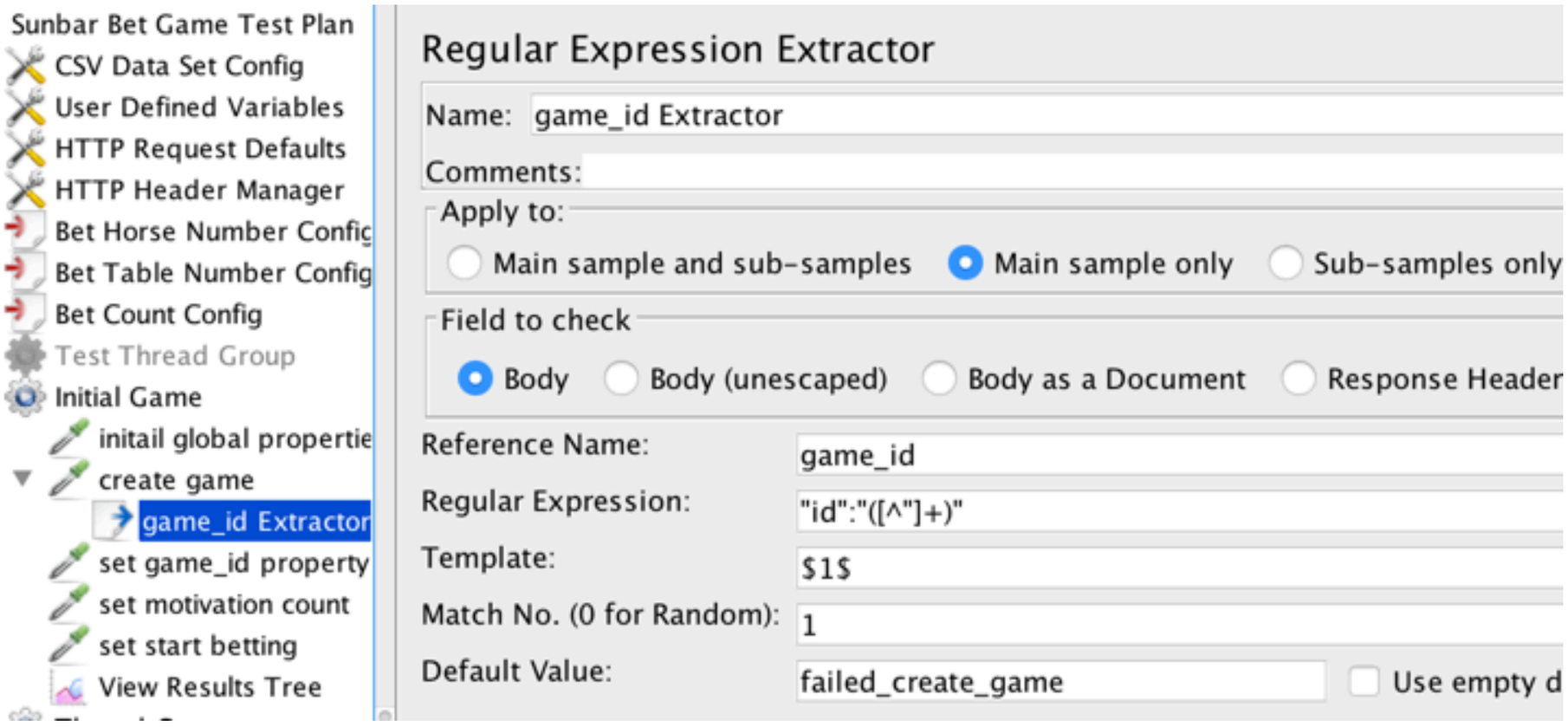
递增：
${__counter(FALSE,charge_order_count)};

判断：
${__javaScript(${charge_order_count} < ${thread_num})}
${__jexl3(${prepare_count} == 1)}

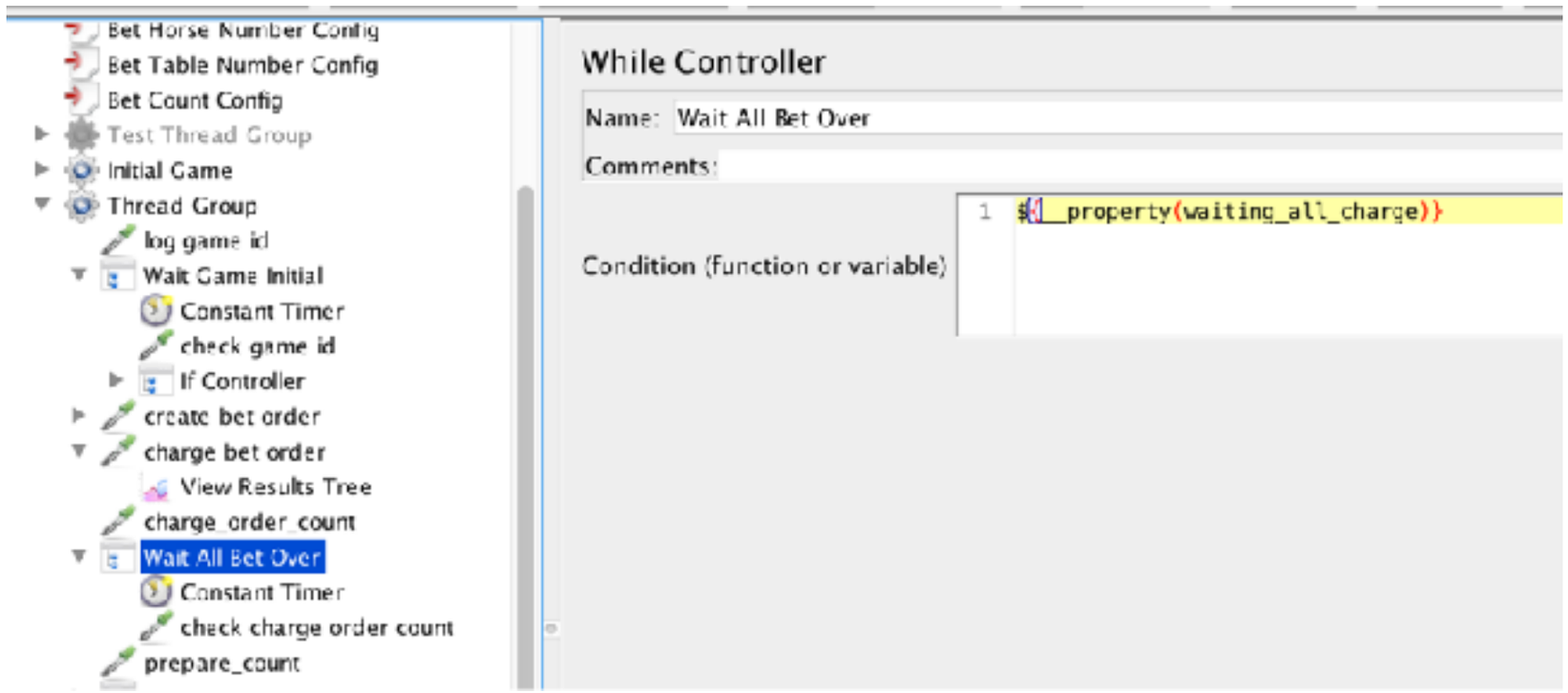
BeanShell

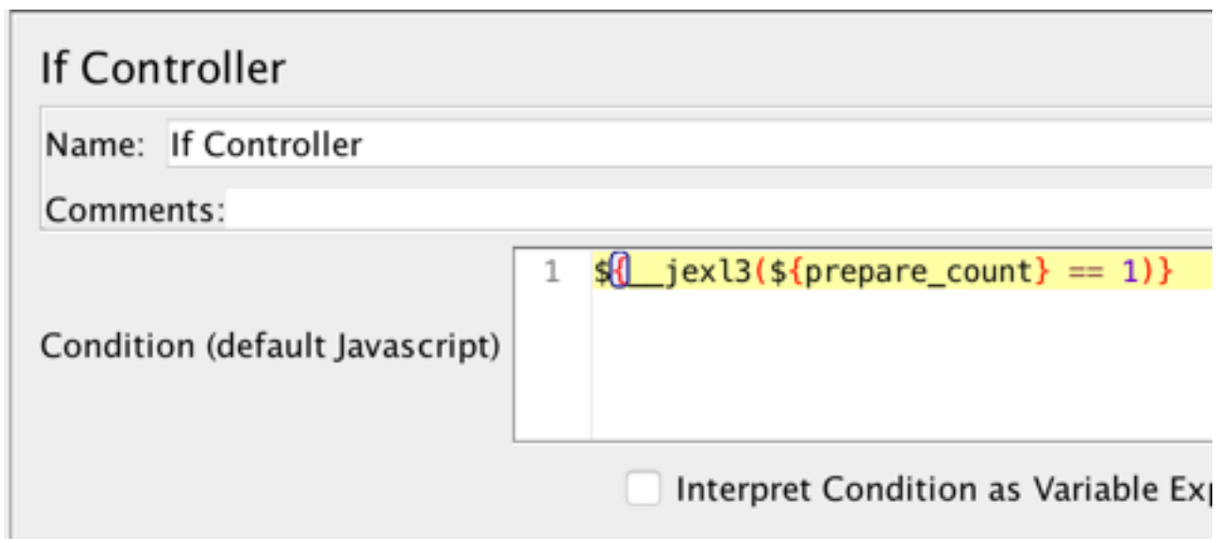
```
int channel_num = ${__javaScript(parseInt(${__threadNum}/1000))};

String channel = "c" + channel_num;

vars.putObject("chat_channel",channel);

log.info("chat channel --> {}",channel);
```

Regular Expression Extractor



Expression


Bet Horse Number Config
Bet Table Number Config
Bet Count Config
Test Thread Group
Initial Game
Thread Group
log game id
Wait Game Initial
Constant Timer
check game id
If Controller
create bet order
charge bet order
View Results Tree
charge_order_count
Wait All Bet Over
Constant Timer
check charge order count
prepare_count
While Controller
Name: Wait All Bet Over
Comments:
Condition (function or variable)
1 ${__property(waiting_all_charge)}


If Controller
Name: If Controller
Comments:
Condition (default Javascript)
1 ${__jexl3(${prepare_count} == 1)}
Interpret Condition as Variable Ex

The 3 Parts of Test Report

- Test Environment (Server & Client)
- Test Plan (Procedure)
- Test Result (Aggregation & TPS)

Install Jmeter

1. Install Jmeter

    https://mirrors.tuna.tsinghua.edu.cn/apache//jmeter/binaries/apache-jmeter-3.3.zip

2. Install Plugin Manager

    https://repo1.maven.org/maven2/kg/apc/jmeter-plugins-manager/

    move the jar to <jmeter_dir>/lib/ext

3. Install Plugin

    reopen jmeter and goto plugin manager plugin manager view, chose plugin to install.

4. 添加自己的java测试客户端

    将sonic_parser_test-1.0-SNAPSHOT.jar放到<jmeter_dir>/lib/ext

Config Jmeter Slave

vi bin/jmeter.properties

server_port=1099
server.rmi.localport=1099

vi /opt/jmeter33/bin/jmeter

HEAP="-Xms512m -Xmx4096m"

vi /opt/jmeter33/bin/jmeter

HEAP="-Xms512m -Xmx4096m"

启动: nohup /opt/jmeter33/bin/jmeter-server &

Config Jmeter Master

vi bin/jmeter.properties

remote_hosts=127.0.0.1,172.16.40.10:1099,172.16.10.11:1099

说明：

1、调度机(master)和执行机(slave)最好分开，由于master需要发送信息给slave并且会接收slave回传回来的测试数据，所以mater自身会有消耗，所以建议单独用一台机器作为mater。

2、参数文件：如果使用csv进行参数化，那么需要把参数文件在每台slave上拷一份且路径需要设置成一样的。

3、每台机器上安装的Jmeter版本和插件最好都一致，否则会出一些意外的问题。

Start Test Plan

单机启动
jmeter -n –t test.jmx -l testresults.jtl

集群方式启动
jmeter -n -t 001.jmx -R 172.16.40.10:1099,172.16.10.11:1099 -l result/003.jtl

-n: It specifies JMeter is to run in non-gui mode
-t: Name of JMX file that contains the Test Plan
-l: Name of JTL(JMeter text logs) file to log results
-R: cluster list
-j: Name of JMeter run log file

Thanks!