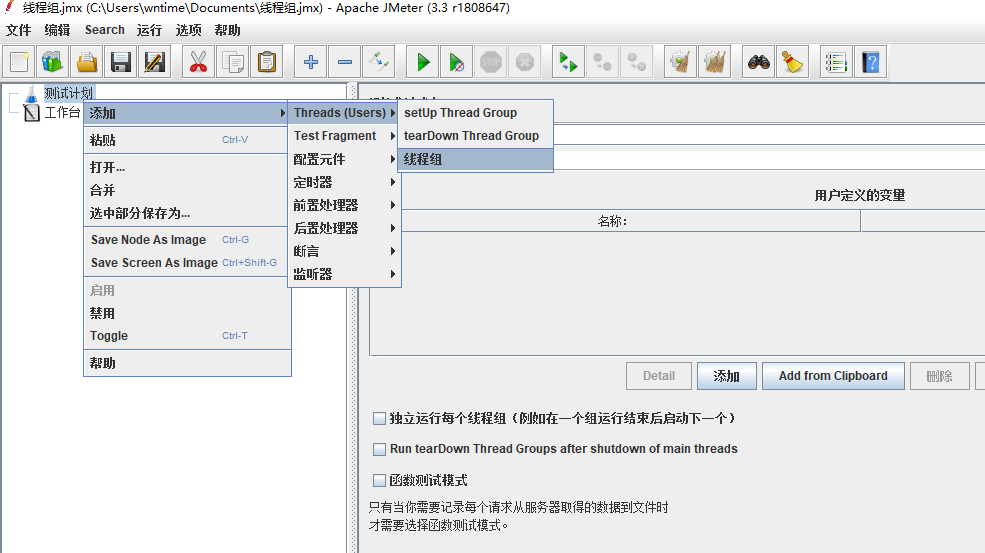
**Gemfire 高并发测试**

在项目中，涉及到对Gemfire 要做高并发测试，尤其是业务高峰期并发量，为达到测试目的，选用jmeter 作为并发测试工具。

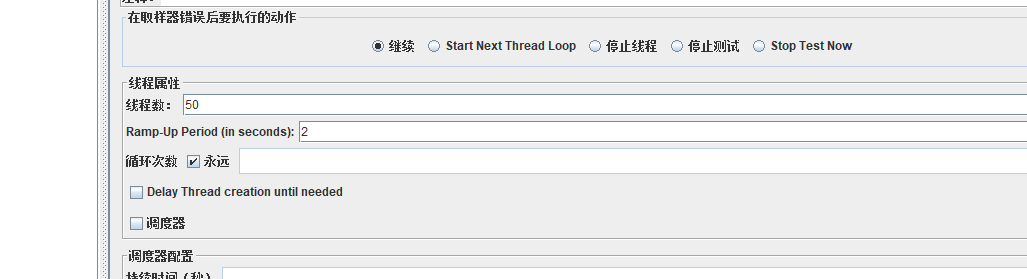
1. 安装Gemfire，省略
2. 安装JDK,要求1.8及以上
3. 安装Jemter

下载地址：<http://jmeter.apache.org/download_jmeter.cgi>

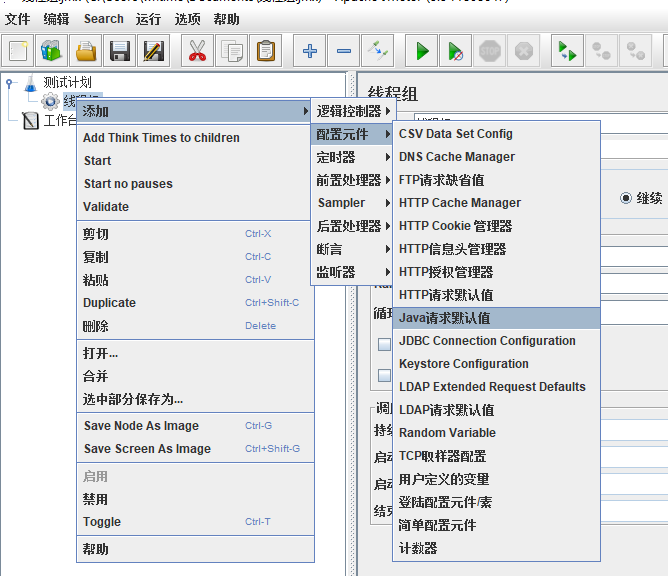
1. 创建线程组:



1. 配置线程



1. 添加配置原件



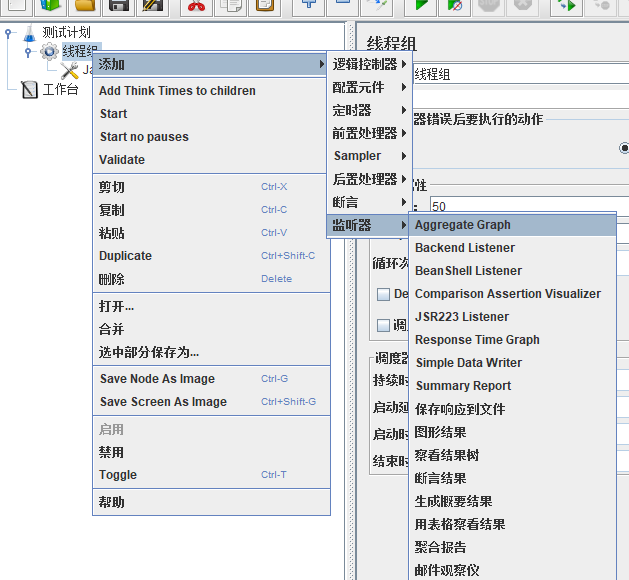
1. 配置gemfire 客户端

利用GemfireClient端并发访问GemfireServer端，GemfireClient代码打包放在jmeter 安装目录lib/ext下。选择jar包，配置客户端请求参数： 端口，region，locator

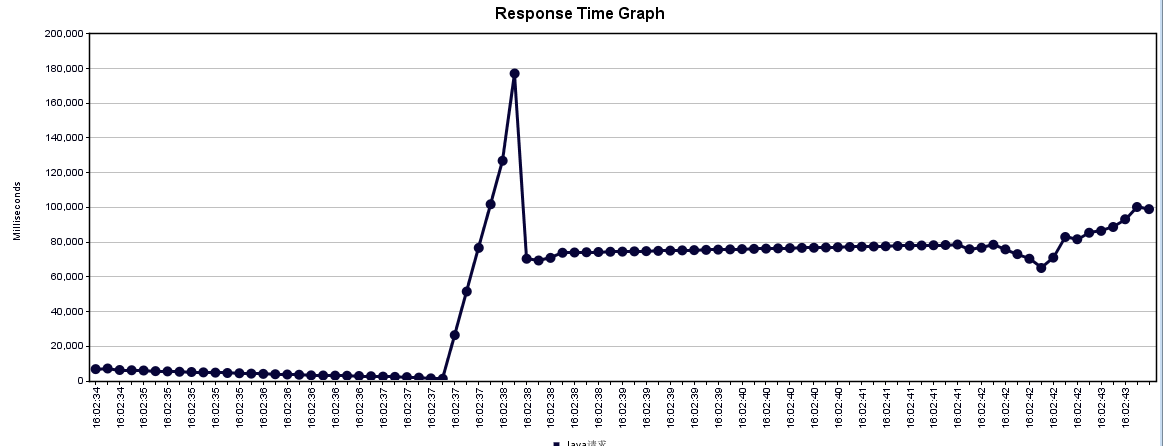




1. 配置监听器



1. 启动jmeter，通过 Response Time Graph 看到响应时间



附：

Gemfire 客户端代码：

public class JMClient

extends AbstractJavaSamplerClient{

private String locator;

private int port;

private String region;

private SampleResult results;

UtilsTools tools = new UtilsTools();

public void setupTest(JavaSamplerContext jsc) {

this.results = new SampleResult();

this.locator = jsc.getParameter("locator", "");

this.port = Integer.parseInt(jsc.getParameter("port", ""));

this.region = jsc.getParameter("region", "");

if ((this.locator != null) && (this.locator.length() > 0)) {

this.results.setSamplerData(this.locator);

}

if ((this.region != null) && (this.region.length() > 0)) {

this.results.setSamplerData(this.region);

}

this.results.setSampleCount(this.port);

System.out.println("init params---------------------------------------port=" + this.port + " locator=" + this.locator + " region=" + this.region);

}

public Arguments getDefaultParameters() {

Arguments params = new Arguments();

params.addArgument("port", "");

params.addArgument("region", "");

params.addArgument("locator", "");

return params;

}

public SampleResult runTest(JavaSamplerContext arg0) {

this.results.sampleStart();

ClientCache cache = new ClientCacheFactory().addPoolLocator(this.locator, this.port).setPoolSubscriptionEnabled(true)

.setPoolSubscriptionRedundancy(1).setPoolReadTimeout(100000).setPdxReadSerialized(true)

.set("log-level", "info").create();

String oql = "select \* from /" + this.region;

QueryService service = cache.getQueryService();

Query query = service.newQuery(oql);

try {

SelectResults<String> result = (SelectResults)query.execute();

String content = "";

for (String value : result) {

content = content + value + "\r\n";

}

this.results.setResponseMessage(content);

this.results.setDataEncoding("UTF-8");

this.results.setResponseData("返回值：" + content);

this.results.setSuccessful(true);

this.results.sampleEnd();

}

catch (Exception e) {

e.printStackTrace();

}

finally {

if (cache != null) {

cache.close();

}

}

return this.results;

}

public void teardownTest(JavaSamplerContext arg0) {}

}