

JMETER 接口性能测试方案

JMETER 简介

JMeter 可以用于测试静态或者动态资源的性能（文件、Servlets、Perl 脚本、java 对象、数据库和查询、ftp 服务器或者其他的资源）。JMeter 用于模拟在服务器、网络或者其他对象上附加高负载以测试他们提供服务的受压能力，或者分析他们提供的服务在不同负载条件下的总性能情况。你可以用 JMeter 提供的图形化界面分析性能指标或者在高负载情况下测试服务器/脚本/对象的行为。

场景介绍

被测 dubbo 服务：

`com.alifi.adapter.service.AdminPortalService`

场景描述：

当以上服务被调用时，服务会去 adminportal 查询对应的数据，并返回结果给测试服务器。

工具原理：

利用 JMETER 调用测试脚本去调用 dubbo 上的服务，并记住每次调用并获取反馈值的响应时间，计算成功率。

JMETER 性能测试环境准备

工具：

- jakarta-jmeter-2.3.2

将 ZIP 压缩包解压缩在你很容易找到的地方，本文放在 C 盘根目录下。

- Eclipse

需要安装 testNG 插件，maven 环境。具体安装步骤可以参考“叶子同学”的 testNG 环境搭建。

JMETER 主要目录结构分析



JMETER 主要的文件夹有:

%Jmeter_home/lib

该文件夹中主要存放测试 JAR 文件运行时需要依赖的 JAR 包.

%Jmeter_home/lib/ext

该文件夹中主要存放带有 main 函数的测试 JAR 文件.只有将测试 JAR 包放在该目录下,才会被 JMETER 工具识别.

后续会有相应的介绍.

JMETER 性能测试步骤及监控

JMETER 做简单的 UI 性能测试及脚本录制:

(<http://www.ibm.com/developerworks/cn/java/l-jmeter/>)

操作很简单.这里不做介绍.

JMETER 做接口性能测试用例步骤:

前提:带有 main 函数的 JAR 包及其运行所以来的 JAR 包都对应的放入 JMETER 文件夹中.

步骤一,新建一个测试用例:

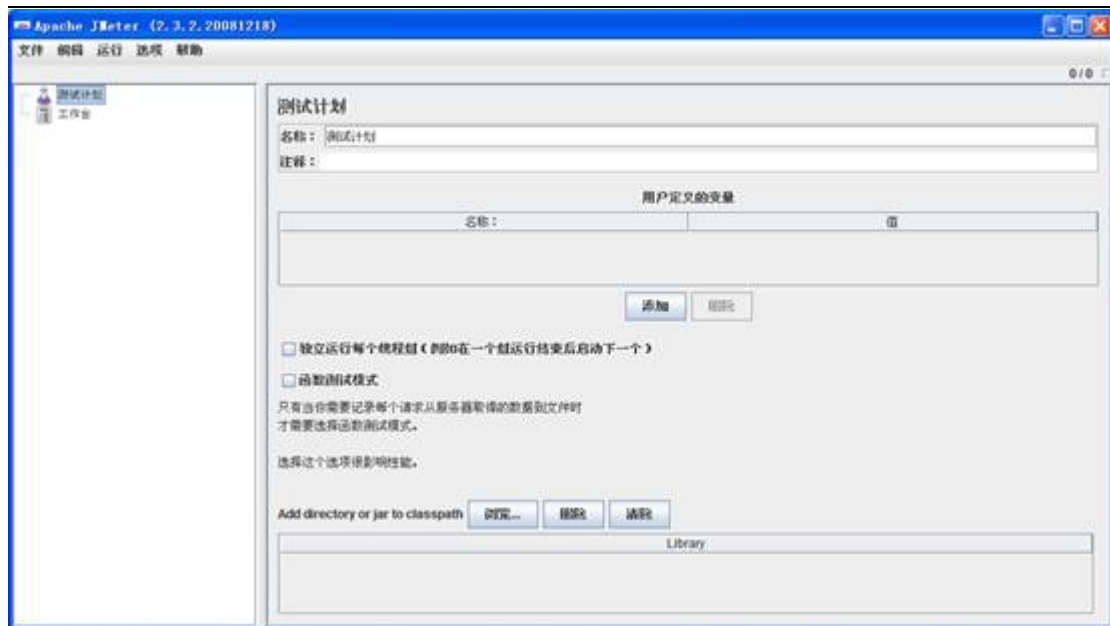
在%Jmeter_home/bin 中找到 Jmeter.bat 文件.双击打开 Jmeter 程序:

软件测试自学与分享平台

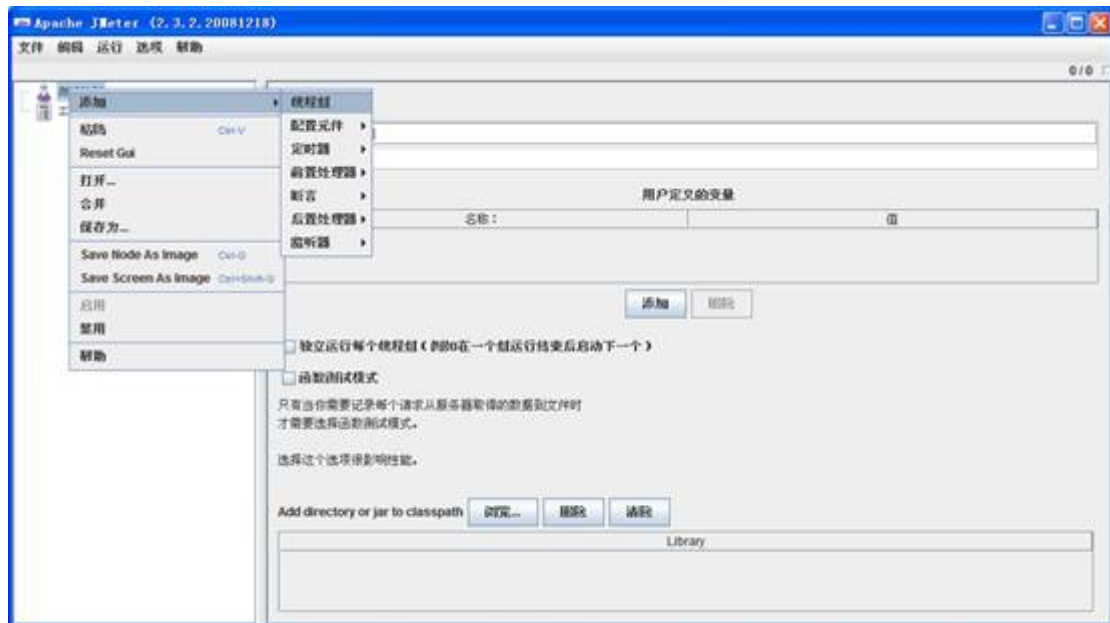
www.besttest.cn

<http://weibo.com/besttest>

性能测试交流群：244735278



在[文件]选项中可以选以往处理过的脚本.现在以新增一个测试用例为例.
步骤二.在[测试计划]中添加一个线程:



右键[测试用例],依次选择[添加]—[线程组];得到如下视图:



线程数(Number of Threads): 设置发送请求的用户数目。

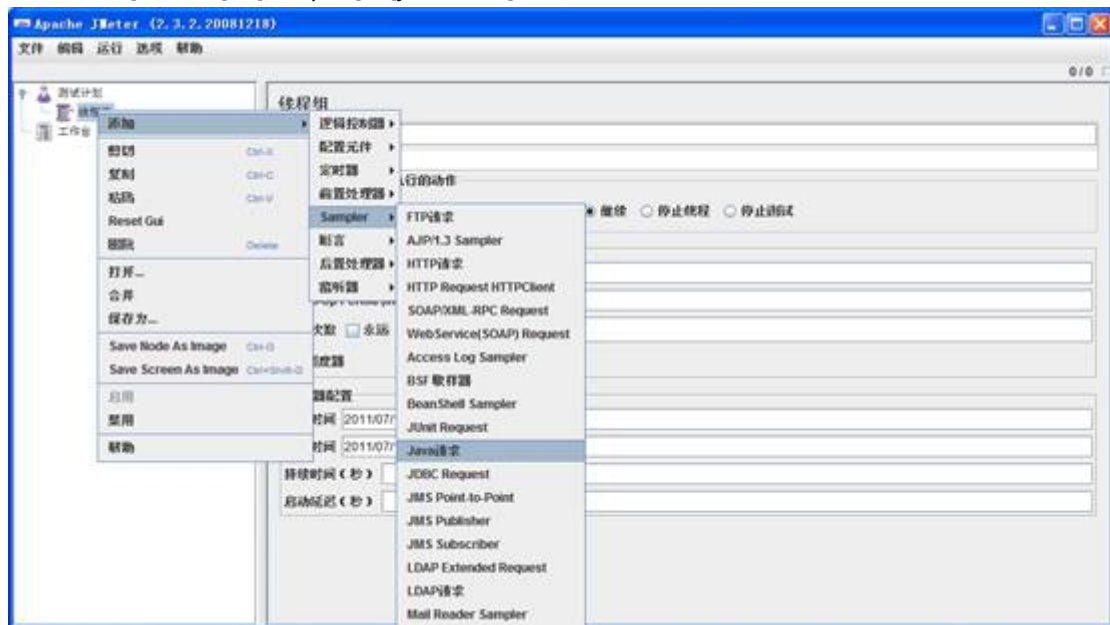
Ramp-up period: 每个请求发生的总时间间隔，单位是秒。比如你的请求数目是 5，而这个参数是 10，那么每个请求之间的间隔就是 $10 / 5$ ，也就是 2 秒。

循环次数(Loop Count): 请求发生的重复次数，如果选择后面的 forever（默认），那么请求将一直继续，如果不选择 forever，而在输入框中输入数字，那么请求将重复指定的次数，如果输入 0，那么请求将执行一次。

在[调度器]中可以选择启动时间,结束时间等。

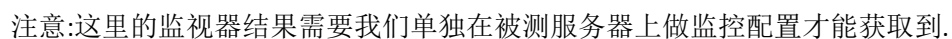
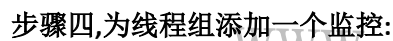
步骤三,在线程组中添加一个动作:

右键[线程组]—[sampler]—[java 请求],如下图:



我们得到一个 JAVA 请求的动作:

性能测试交流群：244735278

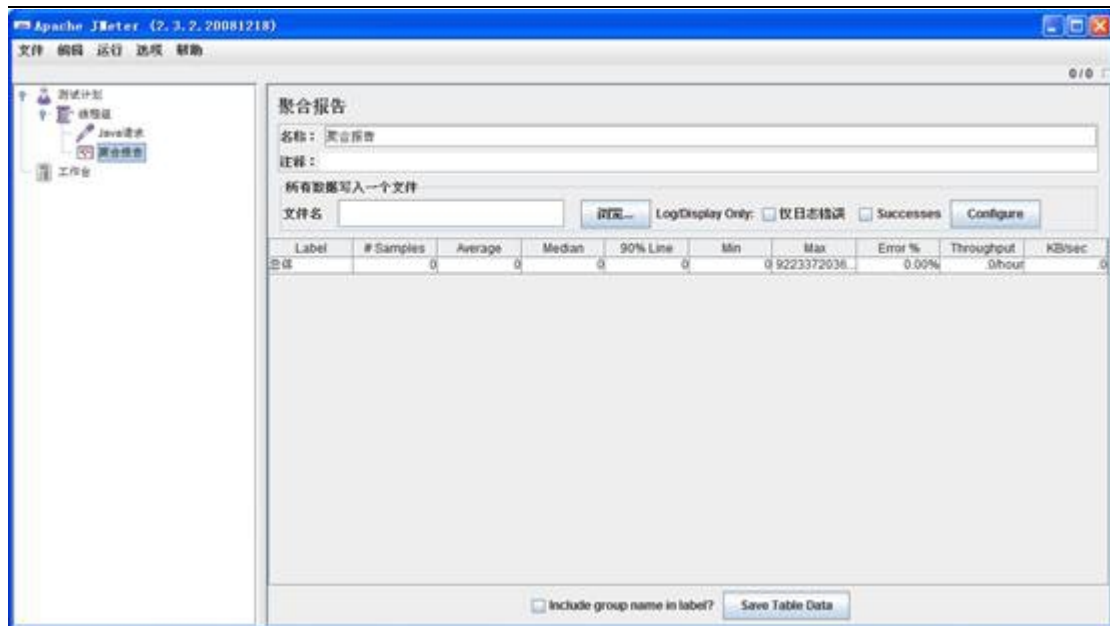


软件测试自学与分享平台

www.besttest.cn

<http://weibo.com/besttest>

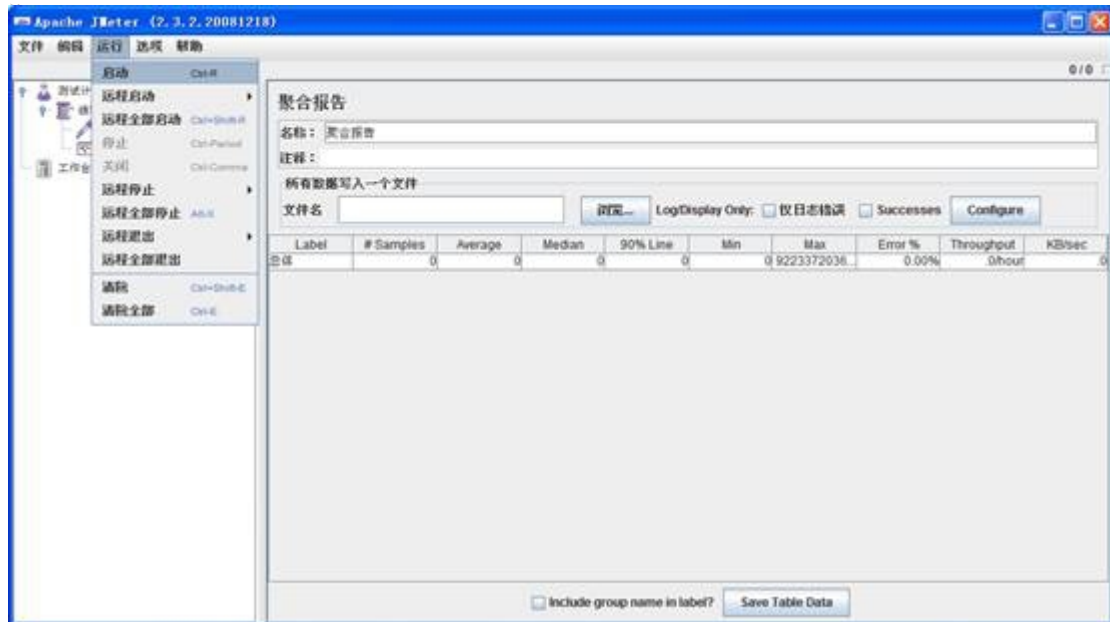
性能测试交流群：244735278



这里可以监控到错误率,最高响应时间,I/O 值.每个监视器的功能不一样.大家可以摸索摸索.

步骤五:运行及结果监控.

注意:为了验证我们的 JAR 是否执行成功,我推荐打开被测目标的日志,检查是否被调用.



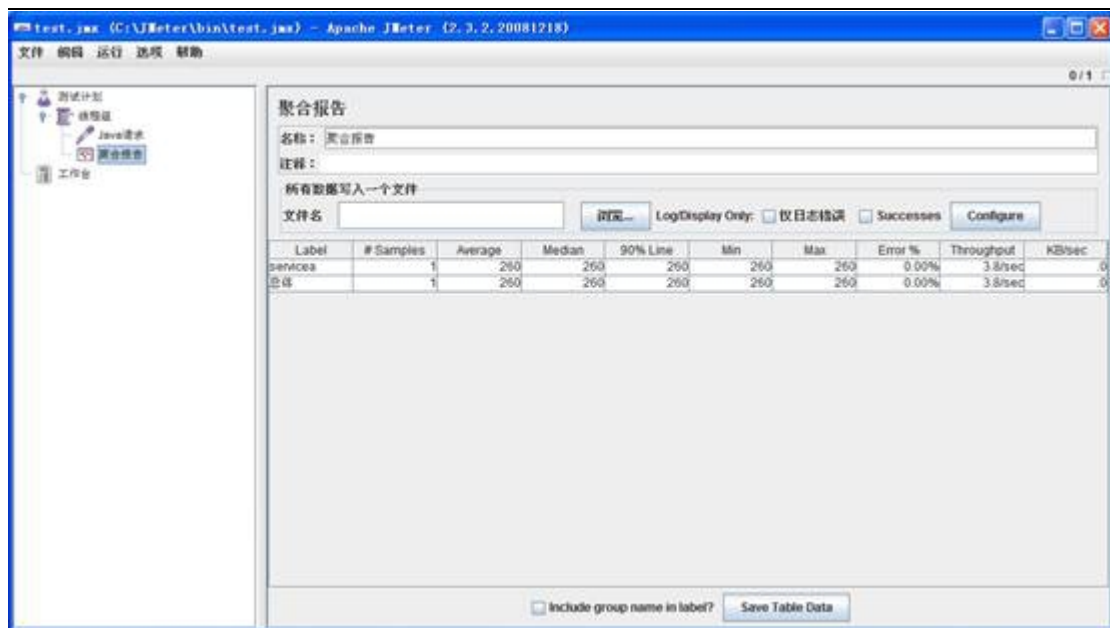
以下是我的简单的运行结果:

软件测试自学与分享平台

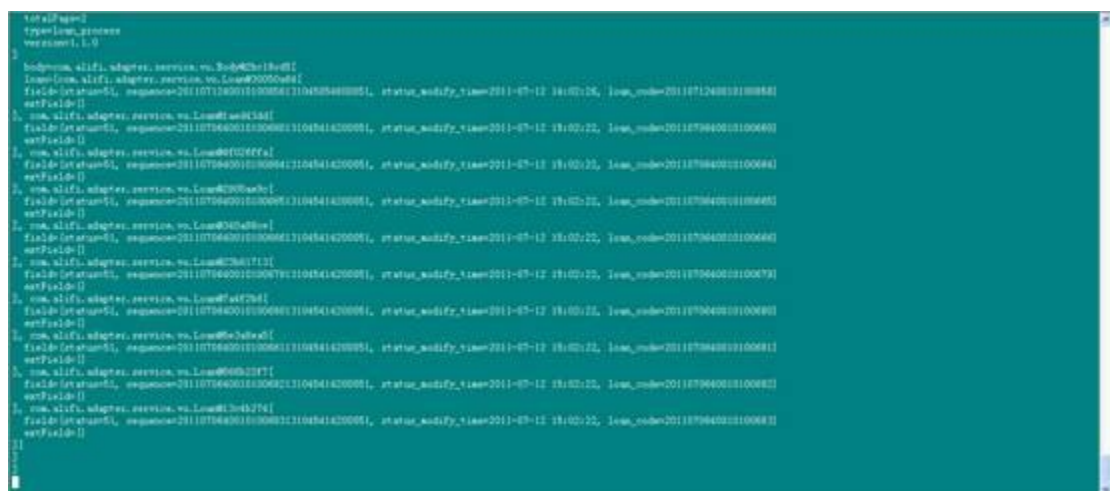
www.besttest.cn

<http://weibo.com/besttest>

性能测试交流群：244735278



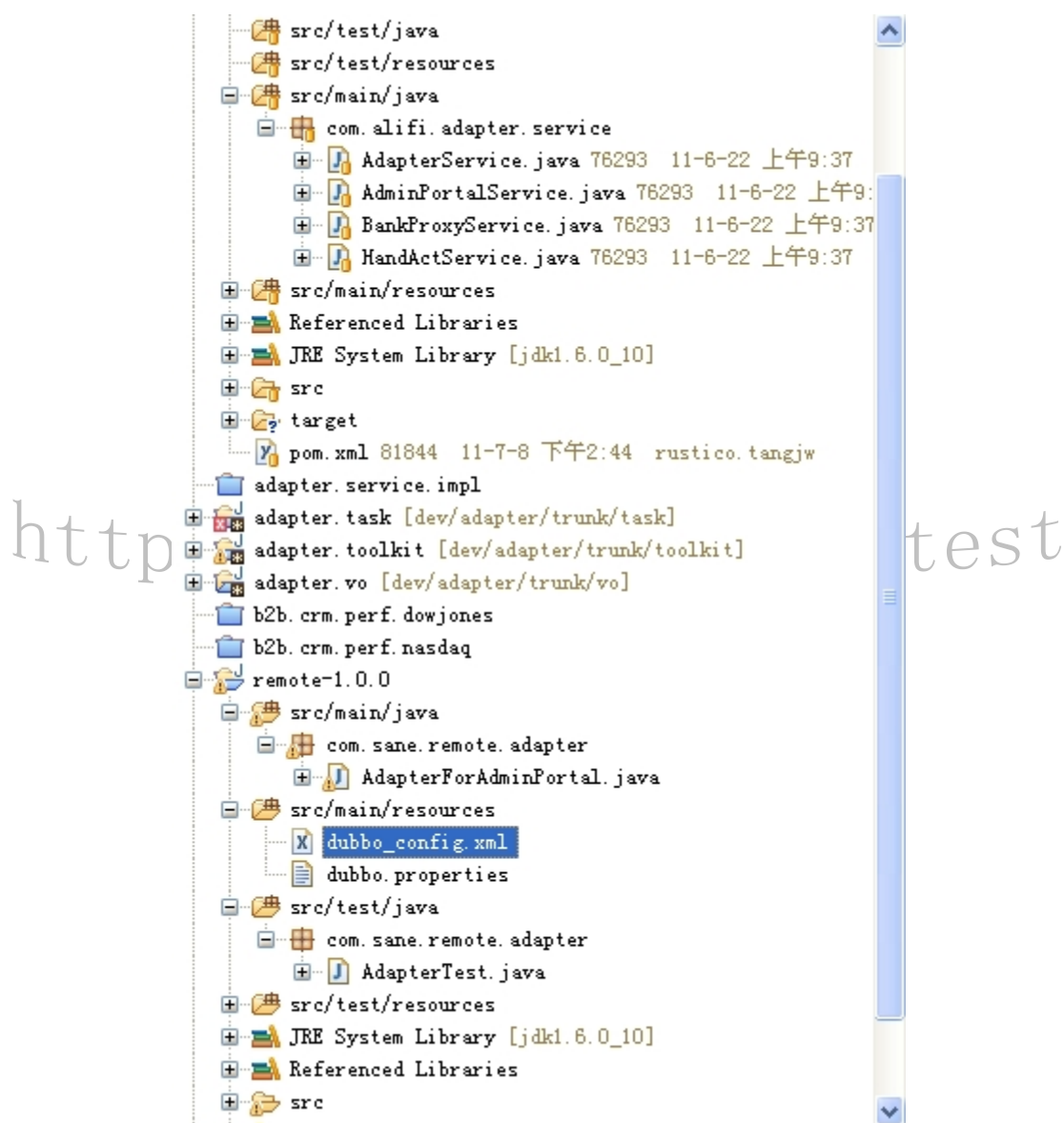
Error 结果为 0%.表明我执行的过程中没有报错.



被测服务器的日志有打印,调用成功.

JMETER 性能测试脚本编写

测试工程目录结构介绍



被测工程:

Adapter 工程,我们主要关注的是

adapter.service.api/src/main/java/com.alifi.adapter.service

里面包含了我们所有注册在 dubbo 上的所有服务.拿 adminPortalService 为例:

```
package com.alifi.adapter.service;

import com.alifi.adapter.service.vo.Data;

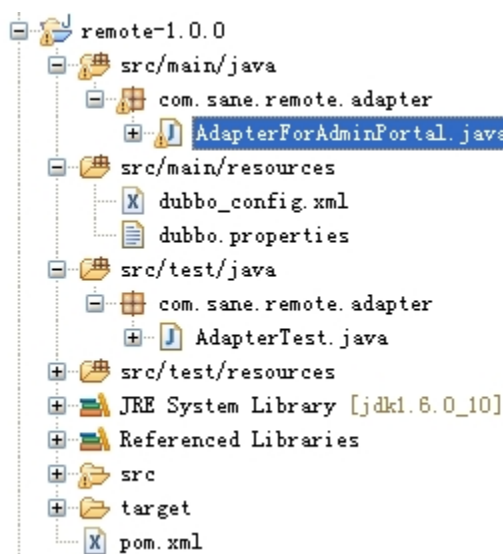
/**
 * Admin Portal Service需要实现的Service. <br>
 * 解决问题: 通过Adapter定义Service由Admin portal实现, 这样Adapter就不用去依赖Admin portal的包
 *
 * @author xubo.wuxb
 */
public interface AdminPortalService {

    /**
     * 获取Admin portal系统的受理进度信息。
     *
     * @param bankCode
     *         银行编号。
     * @param dateStr
     *         日期字符串, 格式为: yyyy-MM-dd。
     * @param page
     *         页码
     * @return 返回Data对象。
     */
    public Result<Data> getLoanProcessStatus(String bankCode, String dateStr, int page);
}
```

我们可以看到整个类需要传入 **String** 类型的 **bankcode,dateStr** 及 **int** 类型的 **page**,并反馈给一个 **Result**.具体的实现方式我们并不关心.

这三个参数决定了我们在测试脚本中不管是引入变量还是写死固定的变量值.

测试工程



Src/main/java 中包含了我们测试 JAVA.

Src/main/resources 中包含了一些我们需要的配置文件;这里我主要存放 dubbo 的配置文件(dubbo.properties)及引入的 dubbo 服务设置(dubbo_config.xml).

Src/test/java 用来存放对一些方法的单元测试(testNG).

POM.XML 文件介绍

配置文件中主要引用的有:

对adapter服务的引用:

```
<dependency>
    <groupId>com.alifi.adapter</groupId>
    <artifactId>adapter.service.api</artifactId>
    <version>1.1.0-SNAPSHOT</version>
</dependency>
```

对 dubbo 的引用

```
<dependency>
    <groupId>com.alibaba.dubbo</groupId>
    <artifactId>dubbo.core.service</artifactId>
    <version>${dubbo-version}</version>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>com.alibaba.dubbo</groupId>
    <artifactId>dubbo.core.registry.default</artifactId>
    <version>${dubbo-version}</version>
    <scope>test</scope>
</dependency>
```

对 Jmeter 的调用:

```
<dependency>
    <groupId>com.alibaba.external</groupId>
    <artifactId>apache.jmeter</artifactId>
    <version>2.3</version>
</dependency>
```

引用单元测试:

```
<dependency>
    <groupId>com.alifi.aliloan.common</groupId>
    <artifactId>common.test</artifactId>
    <version>1.0-SNAPSHOT</version>
    <scope>test</scope>
</dependency>
```

引用 Spring

```
<dependency>
  <groupId>com.alibaba.external</groupId>
  <artifactId>sourceforge.spring</artifactId>
  <version>2.5.6</version>
  <type>jar</type>
</dependency>
```

将依赖包导出项目的方法(mvn clean package -DskipTests=true -Denforcer.skip=true)

```
<build>
  <plugins>
    <plugin>
      <artifactId>maven-dependency-plugin</artifactId>
      <version>2.1</version>
      <executions>
        <execution>
          <id>copy-dependencies</id>
          <phase>prepare-package</phase>
          <goals>
            <goal>copy-dependencies</goal>
          </goals>
        </execution>
      </executions>
      <configuration>
        <includeTypes>jar</includeTypes>
        <overwriteSnapshots>true</overwriteSnapshots>
        <type>jar</type>
        <outputDirectory>D:\remote\lib</outputDirectory>
      </configuration>
    </plugin>
  </plugins>
</build>
```

D:\remote\lib 是我用来存放依赖包的文件夹,用命令打包出来后全部丢入 %Jmeter_home/lib 文件夹中.

命令: (mvn clean package -DskipTests=true -Denforcer.skip=true)

Dubbo 配置文件介绍

Dubbo_config

```
<beans default-autowire="byName"
xmlns="http://www.springframework.org/schema/beans"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:dubbo="http://repo.alibaba-inc.com/schema/dubbo"
xsi:schemaLocation="http://www.springframework.org/schema/beans
http://www.springframework.org/schema/beans/spring-beans-2.0.xsd
http://repo.alibaba-inc.com/schema/dubbo
http://repo.alibaba-inc.com/schema/dubbo/dubbo-component.xsd">

  <dubbo:reference
id="adminPortalService" interface="com.alifi.adapter.service.Admin
PortalService" version="1.0.0" /> <!-- 定义dubbo的服务.-->
</beans>
```

主要是将 `com.alifi.adapter.service.AdminPortalService` 引入 bean 容易
中方便主函数中引用. 大家可以改成自己需要测试的服务目标.

`id="adminPortalService"` 这里的 id 可以自己随便定义, 相当于一个别名.

Dubbo.properties

这个就不过多介绍了, 就是 dubbo 的配置文件. 主要是修改 dubbo 的注册地址及测试服务名称.

```
dubbo.application.name=adapter
dubbo.registry.address=10.249.198.180:9090
dubbo.monitor.address=N/A
dubbo.service.server.port=20880
dubbo.service.console.port=30880
dubbo.service.server.weight=10
dubbo.service.min.thread.pool.size=10
dubbo.service.max.thread.pool.size=300
dubbo.service.thread.keep.alive.seconds=300
dubbo.service.shutdown.wait.seconds=10
dubbo.logging.level=INFO
dubbo.logging.root=/logs
dubbo.logging.file=adapter.log
```

软件测试自学与分享平台

www.besttest.cn

<http://weibo.com/besttest>

性能测试交流群：244735278

JMETER 代码格式及注意事项

所有的代码注意事项已经在脚本中用注释形式全部展现. 请阅读 `adminportalForAadminPortal.java` 源码.

注意事项,在编写完 `main` 函数之后,有必要进行 `testng` 单元测试.

在 `Cmd` 环境下找到工程目录,获取依赖的 `JAR` 文件.和 `Jmeter` 需要调用的带有 `MAIN` 函数的 `JAR` 文件.

JMETER 性能测试心得

关于 `JMETER` 调用接口时候传入参数的方式还没有研究透.

正在努力地在脚本中添加日志结构,方便跟踪到错误信息.

`JMETER` 对接口测试的方便在于,对 `dubbo` 的服务可以配置话,或者架设在 `ci` 上进行可配置话的远程性能测试.

www.besttest.cn
<http://weibo.com/besttest>