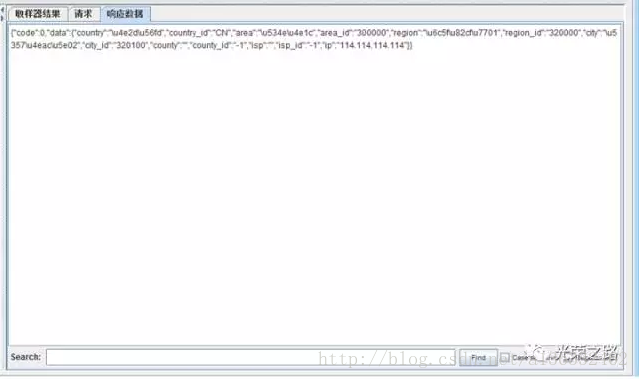
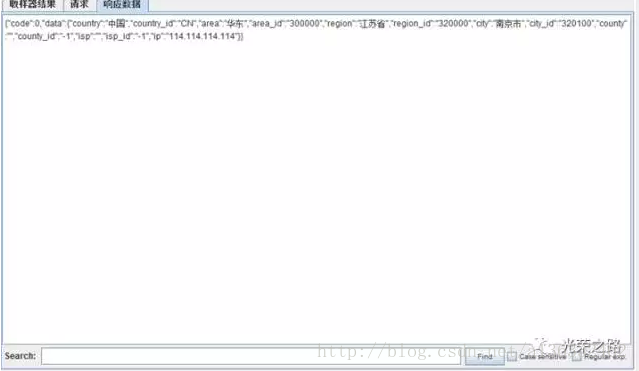
在jmeter工具的使用中，不管是[**测试**](http://lib.csdn.net/base/softwaretest)接口还是调试性能时，查看结果树必不可少，然而在查看响应数据时，其中的中文经常以Unicode的编码形式显示，如图1。这样不能直接查看到对应的中文是否为期望的结果，很不方便。



（样例请求为http://ip.taobao.com/service/getIpInfo.[**PHP**](http://lib.csdn.net/base/php)?ip=114.114.114.114）

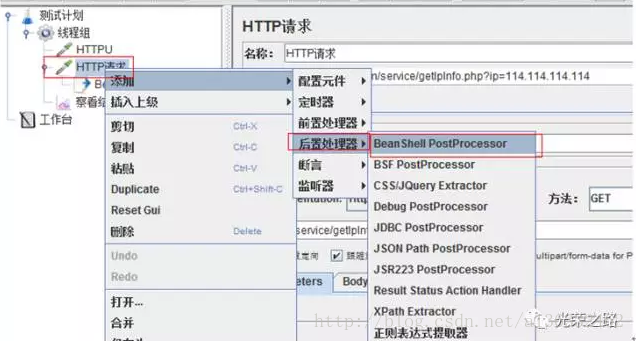
         我找到一个曲线救国的方法，供大家参考。得到结果，如图2。



操作步骤

第一步

在对应请求上添加后置处理器BeanShellPostProcessor，如图3.



得到页面，图4。

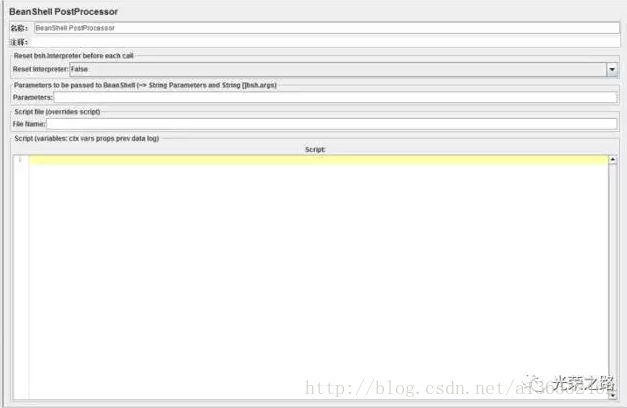
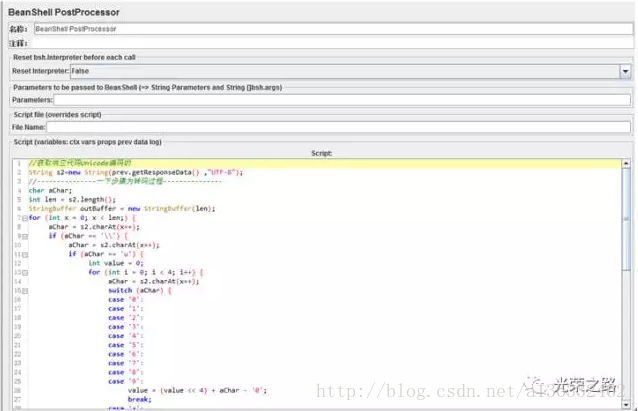


图4

第二步

在后置处理器BeanShellPostProcessor的script中贴入Unicode的转中文的编码，如图5。



第三步

正常调试并查看结果树，发现之前的Unicode的编码的中文，已经转码完毕。

<End>

原理

         通过BeanShell内置变量prev，获得响应数据，经过[**Java**](http://lib.csdn.net/base/javase)程序编码，把Unicode代码转成中文，最后修改查看结果树中响应数据为转换完毕的中文数据。

备注

1、  在性能测试前，请把这个后置处理器删除，不然会大量消耗本机的内存和CPU，影响性能的结果

2、  以上代码中的转码过程可以更改为更适合的代码，但要注意的是Jmeter3.0这个版本依旧对Integer.parseint()方法不能很好支持，运行会报错。

3、  关于获取响应结果的变量值prev，具体API查看：

http://jmeter.apache.org/api/org/apache/jmeter/samplers/SampleResult.html

代码如下：

//获取响应代码Unicode编码的

        String s2=new String(prev.getResponseData(),"UTF-8");  
//---------------一下步骤为转码过程---------------  
        char aChar;  
        int len= s2.length();  
        StringBuffer outBuffer=new StringBuffer(len);  
        for(int x =0; x <len;){  
            aChar= s2.charAt(x++);  
            if(aChar=='\\'){  
                aChar= s2.charAt(x++);  
                if(aChar=='u'){  
                    int value =0;  
                    for(int i=0;i<4;i++){  
                        aChar= s2.charAt(x++);  
                        switch(aChar){  
                            case'0':  
                            case'1':  
                            case'2':  
                            case'3':  
                            case'4':  
                            case'5':  
                            case'6':  
                            case'7':  
                            case'8':  
                            case'9':  
                                value=(value <<4)+aChar-'0';  
                                break;  
                            case'a':  
                            case'b':  
                            case'c':  
                            case'd':  
                            case'e':  
                            case'f':  
                                value=(value <<4)+10+aChar-'a';  
                                break;  
                            case'A':  
                            case'B':  
                            case'C':  
                            case'D':  
                            case'E':  
                            case'F':  
                                value=(value <<4)+10+aChar-'A';  
                                break;  
                            default:  
                                throw new IllegalArgumentException(  
                                        "Malformed   \\uxxxx  encoding.");}}  
                    outBuffer.append((char) value);}else{  
                    if(aChar=='t')  
                        aChar='\t';  
                    else if(aChar=='r')  
                    aChar='\r';  
                    else if(aChar=='n')  
                    aChar='\n';  
                    else if(aChar=='f')  
                    aChar='\f';  
                    outBuffer.append(aChar);}}else  
                outBuffer.append(aChar);}  
//-----------------以上内容为转码过程---------------------------  
//将转成中文的响应结果在查看结果树中显示

        prev.setResponseData(outBuffer.toString());