

# 基于Java的通用WebService的设计

凌 娟

(河北省保定日报社 网络部 河北 保定 071000)

**摘 要:** 为解决软件项目中批处理命令的跨平台问题和批处理命令的远程调用问题, 提出一种基于Java的通用WebService设计方案, 该设计方案能解决批处理命令的跨平台问题和远程调用问题, 同时还具有一定的通用性: 在增加或修改批处理命令时不需要为新增加或修改的命令编写新的WebService。

**关键词:** WebService; java; 通用性; 批处理命令

**中图分类号:** TP3 **文献标识码:** A **文章编号:** 1671-7597 (2010) 0320060-01

## 1 绪论

1.1 问题的提出。以Windows为平台开发的基于Web的软件系统为了完成系统后台的某特殊功能一般都要用到批处理命令, 这样在完成的系统中就会存在一些批处理命令。这些批处理命令的存在会降低系统的可移植性。

WebService技术是建立可互操作的分布式应用程序的新平台, 它具有以下特征: 完好的封装性; 面向消息, 松散耦合; 使用标准协议规范; 高度可集成能力[1]。应用WebService对批处理命令进行包装可以解决上述的跨平台和远程调用两个问题。但应用WebService需要对每个批处理命令进行包装, 为每个批处理命令编写单独的WebService。这样就会带来WebService过多, 难于管理, 客户端难于调用的问题, 并且当添加新的批处理命令或修改已有的批处理命令时需要重新编写WebService。为了解决这个问题, 就需要设计一种通用的WebService来包装所有的批处理命令, 并且当增加或修改批处理命令时不需要对WebService进行增加或修改。

1.2 通用WebService的设计目标。通用WebService的设计目标保护以下几个方面: ① 将批处理实现的功能进行包装, 提高软件的跨平台能力; ② 将批处理实现的功能进行包装, 使客户端能够远程调用这些功能; ③ 通用性, 增加或修改批处理命令时不需要对WebService进行修改。

1.3 通用WebService的设计方案。为了实现上面提出的三个目标, 本文提出了一种基于Java的通用WebService设计方案。该方案能解决使用java语言编写的软件的上述三个问题, 方案如下:

在服务器端编写一个java类, 该类对外提供两个方法: 第一个方法是返回软件中所有的批处理命令列表; 第二个方法是根据传入的批处理命令名称和执行参数执行该批处理命令并将执行结果返回给调用者。在软件中定义一个配置文件, 在该配置文件中将批处理命令执行的类和批处理命令的名称进行映射。当客户端调用服务器端的批处理命令时第二个方法会根据批处理命令的名称到配置文件中查找到相应的类然后执行该类中的方法。

将该类包装成WebService, 并将以上的两个方法提供给客户端调用。这样就解决了上述的三个问题。

## 2 通用WebService工作原理

客户端调用WebService提供的第一个方法得到服务器端的批处理命令列表, 然后调用第二个方法来执行批处理命令并得到批处理命令的执行结果。客户端也可以直接调用第二个方法来执行已知的批处理命令。通用WebService的工作流程图如图2-1所示:

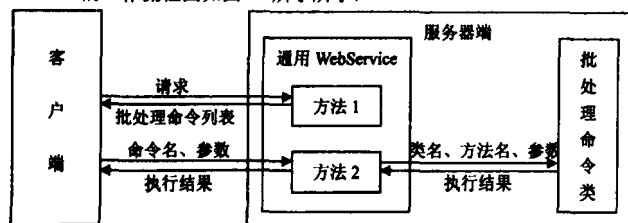


图2-1 通用WebService工作流程图

## 3 关键技术分析

3.1 通用性。本文所指的通用性是指WebService将服务器端的所有批处理命令进行统一包装而不必为每一个批处理命令编写单独的WebService。通用WebService通过对外提供的第二个方法来实现通用性。

通用WebService对外提供的第二个方法接受两个参数: 需要执行的批处理命令的名称和执行参数。该方法通过查找服务器端的批处理命令配置文件找到该批处理命令要执行的类和该类中的方法。根据java反射机制生成该类的一个实例, 调用该类中需要执行的方法并传入执行参数来实现批处理命令实现的功能。

3.2 批处理命令配置文件设计。批处理命令配置文件用来统一管理系统中向外提供的批处理命令, 系统中向外提供的批处理命令都要在批处理命令配置文件进行定义。通用WebService中的方法在该配置文件中通过批处理命令的名称查找需要执行的类和方法, 所以批处理命令配置文件中的命令定义要包含命令名称、执行类、执行方法, 其中命令名称必须唯一。批处理命令的执行参数都是字符串, 所以在配置文件中不必定义执行参数的类型。另外在配置文件中可以定义命令参数个数以便在命令执行前进行参数个数检查, 定义命令解释以便使调用端对要调用的批处理命令有一个直观的理解。

为了增强系统的可移植性, 批处理命令配置文件采用XML格式。

3.3 命令执行结果返回给调用端。服务器端的批处理命令的执行结果都是输出到服务器端的标准输出设备, 而客户端调用服务器端的批处理命令需要将执行结果返回, 应用java I/O重定向技术可以解决这个问题。

先将输出到服务器端标准输出设备的输出流重定向到一个字符串, 然后执行命令, 执行完毕后再将输出流重定向回服务器端的标准输出设备。将存放有命令执行信息的字符串返回给调用端。

3.4 异常定义。异常在程序的执行过程中是不可避免的, 如果服务器端程序执行发生异常则需要把异常信息返回给调用端, 使调用端知道异常的信息。

该通用WebService在服务器端发生的异常主要包括以下四类: 异常1: 客户端要调用的批处理命令在服务器端不存在; 异常2: 客户端要调用的批处理命令在服务器端不存在相应的类; 异常3: 客户端要调用的批处理命令在服务器端不存在相应的方法; 异常4: 客户端调用批处理命令时传入的参数个数不对。

定义四类异常Exception e1、e2、e3、e4分别代表以上四类异常, 定义异常Exception e代表其他异常。定义五个字符串代表五种异常信息: ① “no cmd”; ② “no class”; ③ “no method”; ④ “parameter wrong number”; ⑤ “other exception”。

当服务器端程序发生异常时需要将异常捕获并将异常信息返回给调用端。

## 4 总结与展望

本文所述的通用WebService已经在某国有大型企业的绩效考核信息系统中得到了使用。由于其结构简单, 使用方便, 通用性强等有点, 在实际使用中得到了较好的效果。该通用WebService的使用方便了用户在远端对系统的管理, 也使绩效考核系统能平滑的由原来的Windows平台过渡到IBM AIX平台, 使批处理命令不再是系统进行平台移植的阻碍。