

一种分布式系统自动升级的方法^①

赵珂¹ 郑尚辉² 沈洪骥² (1.山东教育学院 计算机科学与技术系 山东 济南 250013;
2.山东新华书店集团科技公司 山东 济南 250002)

摘要: 介绍了利用 PowerBuilder 提供的技术实现版本控制以及自动升级版本功能的方法, 给出了自动升级流程和关键技术的实现, 提出的升级方案由于基于 Web Service, 所以具有普遍性, 能够适合分布式系统架构下开发的软件的自动升级。目前该自动升级功能已在书店的物流信息管理平台中得到良好的应用。

关键词: PowerBuilder ; Web Service ; 分布式系统; 自动升级

An Approach for Auto-Update Distributional System

ZHAO Ke¹, ZHENG Shang-Hui², SHEN Hong-Ji²

(1.Department of Computer, Shandong Education Institute, Jinan 250013, China;

2.Technology Company, Shandong Xinhua Bookstore, Jinan 250002, China)

Abstract: Available technology has been used on a program that contains an automatic upgrade feature called PowerBuilder version control. This paper presents the upgraded program, since it is based on Web Service, it is suitable for a distributional system developed under the framework of automatic upgrades. At present, the automatic update feature has been used in bookstores as a dependable and logistical information management platform.

Keywords: PowerBuilder; Web service; distributional system; auto-update keyword

1 引言

在分布式系统的开发中, 一个困扰开发者的问题就是如何保持客户端软件的一致性, 即开发方对系统的客户端软件进行了修改和补充, 也就是系统的版本发生了改变, 那么就要对客户端软件进行更新, 防止出现新老版本不统一的问题^[1,2]。如果要开发方亲自对每一个客户端的软件进行版本更新的话, 势必增加开发方的负担。如果系统能够自动进行版本检测与版本更新功能, 那么将给用户和开发方都带来极大的方便。本文主要介绍了 powerbuilder11 实现分布式系统的版本检测与版本自动更新功能^[3]。

2 设计思路及关键技术

2.1 应用背景

书店物流信息管理平台实现了教材到货、配发、

包装、发运的自动化和退货配送一体化。该平台采用了分布式应用程序体系结构, 中间层服务 Web Service 完成绝大多数的数据处理和数据库访问, 客户端通过调用运行在服务器上的组件或对象的方法, 来使用服务器提供的服务。版本检测与版本功能属于物流信息管理平台的一个子模块, 用来处理由于业务的变化所进行的程序改动的更新。

2.2 设计思路

用户登陆物流信息管理平台时触发自动升级程序, 客户端获取服务器端升级文件的版本号, 当服务器端版本比客户端版本高时, 获得一个最新版本的客户端文件列表(有版本等信息), 然后和本地的文件列表对比, 得到一个需要更新的文件列表, 然后对这个需要更新得列表中的文件逐个更新, 替换掉客户端原来的文件, 最后重新启动客户端。替换成功后, 则更新

^① 收稿时间:2010-01-25;收到修改稿时间:2010-03-24

本地版本号，使软件版本保持同步。因此，当进行维护时，只需替换修改过的文件列表即可，无需全部文件替换，这样网络传输代价会小很多，自动升级的效率也会较高。

客户端的版本号在 pb 程序中的本地文件存储，开发人员修改应用程序的某些功能并且更新本地文件的版本号，编译为可执行文件及 pb 动态库文件，然后修改文件成功上传至服务器。服务器端的文件版本号通过在系统的数据库中建立 t_version 表 update 字段进行控制。表结构如下：

表 1 t_version 字段说明

字段名称	类型	含义
version	Varchar2(4)	版本号
filename	Varchar2(255)	文件名
path	Varchar2(1024)	路径

对自动升级程序处理流程设计如下：

(1) 将需要升级的文件上传存储在服务器上。

(2) 用户登陆物流信息管理平台时触发自动升级程序，检查程序中客户端版本号是否比服务器端对应文件的版本号小。如果小就意味着需要更新，将获取可升级文件列表并启动自动下载程序，下载完成后替换客户端对应的文件。

(3) 如果下载成功即关闭自动升级程序，并重新启动客户端应用程序。

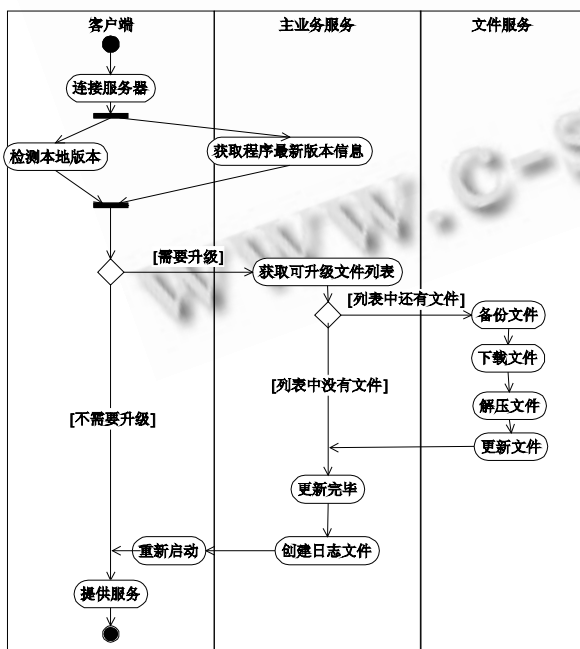


图 1 系统活动图

2.3 关键技术

2.3.1 Web Service 分布式计算模式

分布式计算方式提供了一种将应用程序的用户界面部件与应用程序所需的业务逻辑进行分离的自然途径。通过把业务逻辑集中在中间层服务器上，可以降低客户端的工作负载，控制对敏感信息的访问。

Web Service 是目前程序设计领域中的一项新技术，是一个崭新的分布式计算模式。Web Service 是指在 Internet/Intranet 上描述、发布、查找和调用的自包含的组件技术及相关的协议和标准，它逻辑性的为其他应用程序提供数据与服务^[4,5]。各应用程序通过网络协议和规定的一些标准数据格式(HTTP, XML, SOAP)来访问 Web Service，通过 Web Service 内部执行得到所需结果。一旦部署以后，其他 Web Service 应用程序可以发现并调用它部署的服务。

2.3.2 压缩和解压技术

为了节省空间、提高升级速度，减少客户端动态升级系统与服务器之间的数据流量，采用了压缩和解压算法。中间层服务将服务器端升级程序文件压缩成通用的.zip 文件，客户端程序下载自动识别，解压。WebService 中自己定义的将下载文件压缩的服务 of_download_file(string as_file_name, ref blob ablb_data, ref string as_errorbuffer)。

功能：将需要下载的文件(不包含路径)as_file_name 压缩成二进制数据 ablb_data，首先判断文件是否存在，如果存在的话，实例化文件流，读取文件到二进制数组中，调用 of_compress 方法进行内存压缩：

```
n_zipclass Inv_zip
Inv_zip.of_compress(ablb_data, lbl_temp)
```

3 动升级的实现

该实现方法主要包括 3 个过程：一是开发人员或维护人员将要升级的文件通过 SVN 版本控制软件上传到服务器中，并且把文件新版本号等信息存入 version 表；二是用户进入主程序前的版本检测；三是用户进入主程序前的版本自动更新。

3.1 程序上传

PB 编译应用程序后生成动态库的形式，包括 PBD 和 DLL，动态库中包含了相应 PB 库文件中所有编译好的代码^[6]。将系统按照功能划分成不同的模块，每

一个模块对应于一个 PB 应用库文件,不但容易划分应用程序,方便了应用程序的管理工作,而且增加了系统的可维护性。开发人员修改了应用程序的某些功能后无须重新发布整个应用程序或其他文件,只需发布包含该功能的动态库即可,将要升级的动态库文件通过 SVN 上传发布到服务器中。

3.2 版本检测

用户登录系统时会自动校验版本号,用户程序的版本检测过程如下:登陆系统时,客户端启动,通过函数 of_getversion()得到客户端本地文件的版本号,服务器端则是通过调用 Webservice 中自己定义的 of_get_server_version 服务来实现。检查服务器端版本号是否高于本地版本号,如果高于则启动版本自动更新程序 upp_version.exe。

3.3 版本下载

版本下载是版本检测与版本自动更新的重要环节,版本下载更新的顺序是:备份压缩文件—下载文件—解压文件—更新文件。

根据升级文件程序列表,将本地需要升级的程序文件通过 SVN 存放在服务器的指定目录中,客户端创建旧版本备份目录以及新版本下载备份目录(创建的时候清除之前目录中存在的文件),调用文件传输(webbservice 中定义的 of_download_file 服务)将指定的压缩文件下载到本地新版本下载备份目录中,每下载一个文件,调用解压缩模块对文件进行格式校验,进行解压缩,并调整升级显示进度,下载成功后,将本地正式程序的旧版本移动到旧版本备份目录中,并将新版本下载备份目录中的程序复制到本地正式程序中。如升级不成功,则利用已备份的旧版本文件进行恢复操作。以上任何步骤出错,则调用异常处理进行处理,退出动态升级系统。以如何使用服务器端的文件替换本地的文件为例,说明其程序实现过程。

```
// 下载新程序到本地临时目录,
Inv_file.of_filewrite 定义的中间层服务
```

```
Inv_file.of_filewrite( ls_save_file_path + "\"
+ as_app_update_filename[i], lbl_file_source)
...
```

```
//把旧版本文件 move 到旧版本目录, 下载新版本文件 move 到本地正式的程序目录下
```

```
For i = 1 to li_down_count
  ls_filename =as_app_update_filename[i]
  FileMove(ls_pro_file_path +'\ ' +
ls_filename, ls_file_bak_path +'\ ' + ls_filename)
  ...//省略解压缩处理
  li_sts = FileMove (ls_save_file_path +'\ ' +
ls_filename, ls_pro_file_path +'\ ' + ls_filename)
  IF li_sts <> Success Then
    gotoError()
  End IF
Next
//更新成功后, 启动客户端
MessageBox("提示","文件更新成功")
Run(ls_pro_file_path + "\物流信息管理系统.exe")
```

4 结语

自动升级功能已成为目前软件系统不可缺少的功能,本文以分布式系统的开发模式为研究对象,提出具体的设计方案及实现方法,对软件系统的自动升级功能的研制与开发,具有实用的参考价值。

参考文献

- 1 杨继家,张丽静,张晓蕾.面向 C/S 模式下的客户端软件自动升级的实现.微计算机应用, 2005,14(3):290-293.
- 2 乌云高娃.动态升级在 MIS 系统中的实现与应用.计算机工程与设计, 2005,12(10):2854-2856.
- 3 叶利华,陶宏才,梁田.基于 COM 的软件在线升级技术.成都信息工程学院学报, 2005,18(1):73-75.
- 4 柴晓路,梁宇奇.Web Services 技术、架构和应用.北京:电子工业出版社, 2003.4-98.
- 5 王莉莉,王力生.Web Service 技术下的企业应用集成.计算机与现代化, 2003,13(4):39-41.
- 6 陈永强.PowerBuilder 编程技术全接触.北京:清华大学出版社, 2007.5-216.