<自动版本更新管理软件>

概要设计说明书

Version <1.1>

目录

[1. 导言 3](#_Toc513621587)

[1.1 目的 3](#_Toc513621588)

[1.2 范围 3](#_Toc513621589)

[1.3 缩写说明 3](#_Toc513621590)

[1.4 术语定义 3](#_Toc513621591)

[1.5 参考资料 4](#_Toc513621592)

[1.6 版本更新信息 4](#_Toc513621593)

[2. 系统分析 4](#_Toc513621594)

[3. 总体逻辑框架结构 4](#_Toc513621595)

[4. 总体设计 5](#_Toc513621596)

[4.1 系统运行环境 6](#_Toc513621597)

[硬件平台： 6](#_Toc513621598)

[软件环境 6](#_Toc513621599)

[网络环境 6](#_Toc513621600)

[5. 界面设计 6](#_Toc513621601)

[6. 数据结构设计 7](#_Toc513621602)

[6.1 逻辑结构设计 7](#_Toc513621603)

[6.2 物理结构设计 7](#_Toc513621604)

[7. 模块设计： 7](#_Toc513621605)

[7.1模块功能： 8](#_Toc513621606)

[7.1.1 获取远端配置文件模块 8](#_Toc513621607)

[7.1.2 设置远端URL 8](#_Toc513621608)

[7.1.3 配置文件管理模块 8](#_Toc513621609)

[7.1.4 查看配置文件信息模块 8](#_Toc513621610)

[7.1.5 修改目标文件更新方式 8](#_Toc513621611)

[7.1.6 新增目标文件 8](#_Toc513621612)

[7.1.7 删除目标文件 8](#_Toc513621613)

[7.1.8 发布版本 8](#_Toc513621614)

[7.1.9 新增配置文件 9](#_Toc513621615)

[7.1.10 删除配置文件 9](#_Toc513621616)

[7.1.11 更新模块 9](#_Toc513621617)

[7.1.12 日志记录模块 9](#_Toc513621618)

[8.异常处理机制： 9](#_Toc513621619)

[8.1异常检测机制： 9](#_Toc513621620)

[8.2异常处理机制： 9](#_Toc513621621)

[8.2.1 出错信息: 9](#_Toc513621622)

[8.2.2 措施 9](#_Toc513621623)

[9. 日志记录规范： 10](#_Toc513621624)

[9.1 日志库 10](#_Toc513621625)

[9.2 日志等级划分： 10](#_Toc513621626)

# 导言

## 1.1 目的

该文档的目的是描述自动版本更新软件项目的概要设计，其主要内容包括:

* + 系统功能间接；
  + 系统结构设计；
  + 系统接口设计；
  + 数据设计；
  + 模块设计；
  + 界面设计。

本文档根据**自动版本更新软件需求规格说明书**，对自动版本更新软件的架构进行说明，为后继的详细设计等工作提供参考和依据。

本文档的预期读者是：

* 设计人员；
* 开发人员；
* 项目管理人员；
* 测试人员。

## 1.2 范围

该文档定义了系统的结构和单元接口，但未确定单元的实现方法，这部分内容将在详细设计/实现中确定。

## 1.3 缩写说明

* OAUS: OS automatic upgrade system(OS自动升级系统)

## 1.4 术语定义

|  |  |  |
| --- | --- | --- |
| 编号 | 术语名称 | 含义说明 |
| 1 | 整体更新 | 软件直接下载一个安装包文件，直接进行覆盖安装 |
| 2 | 部分更新 | 软件只需下载部分文件，重启后即可实现。有些部分更新比如更新数据无需重启，只需重新加载就可以了。 |
| 3 | 面向对象分析 | 面向对象分析(object-oriented analysis)一种新的系统分析方法.它是采用面向对象的风格进行系统分析和需求定义的重要方法.面向对象分析的关键在于理解问题空间并将其模型化。 |

## 1.5 参考资料

[1]韩万江等. 软件工程案例教程：软件项目开发实践（第二版）.机械工业出版社，2011.

[2]张旭. 软件自动更新升级检测系统的设计与实现[J].大连海事大学，2015.

[3]宋凌怡. 软件通用自动升级方法的研究与实现[J].安顺学院学报. 2011(03)

[4]自动版本更新软件需求规格说明书 V2.0

## 1.6 版本更新信息

任何以此版本创建或维护更新，都要追加一条记录。一个版本创建只有一次，但对它的维护更新可能有多次。大版本升级一次，定义为创建一次，如V1.0到V2.0。而V1.0到V1.1，只是维护更新一次。版本更新记录格式，如表1-1所示。版本更新记录如下表所示：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本号 | 创建者 | 创建日期 | 维护者 | 维护日期 | 维护纪要 |
| V1.0 | 第五组 | 2018/5/2 | — | — | — |
| V1.1 | — | — | 第五组 | 2018/5/8 | 增加异常处理机制和日志记录规范的内容 |

# 系统分析

本系统可以实现软件自动版本更新，包括访问远程网站自动获取更新，自动按更新方式更新（包括整体覆盖更新，部分更新），生成版本配置文件，支持将版本配置文件和待更新的目标文件复制到某个目录，生成新的版本。

使用该系统的用户分为两大类，分别为普通用户和管理员部分。管理员可以复制配置文件和待更新的目标文件到网站某个文件夹下；用户可以对软件进行管理和更新，同时支持修改默认网站URL。

# 总体逻辑框架结构

根据系统分析结构，该系统从结构上应满足：

* 桌面应用程序，便于用户操作使用；
* 采用MVC的三层体系结构，分化各个功能组件；
* 采用配置文件信息进行数据的保存；

根据以上分析，本系统按照功能层次可划分为展现层、后台核心组件层、接口层，具体逻辑框架见下页图1总体逻辑框架图所示。

# 总体设计

系统的总体结构设计遵循如下原则：

1. 系统应具有良好的适应性。能适应用户对系统的软件环境、管理内容、模式和界面的要求；
2. 系统应具有可靠性。采用成熟的技术方法和软件开发平台，以保证在以后的实际应用中安全、可靠；
3. 系统应具有较好的安全性。应提供完善的安全机制。
4. 系统应具有良好的可维护性。系统应易于维护、安装；
5. 系统应具有良好的可扩展性。系统应适应未来信息化建设的要求，能方便地进行功能扩展，以建立完善的信息集成管理体系；
6. 系统的设计开发应符合信息安全化建设的要求，以方便实现与其他设备以及各类应用系统的集成。

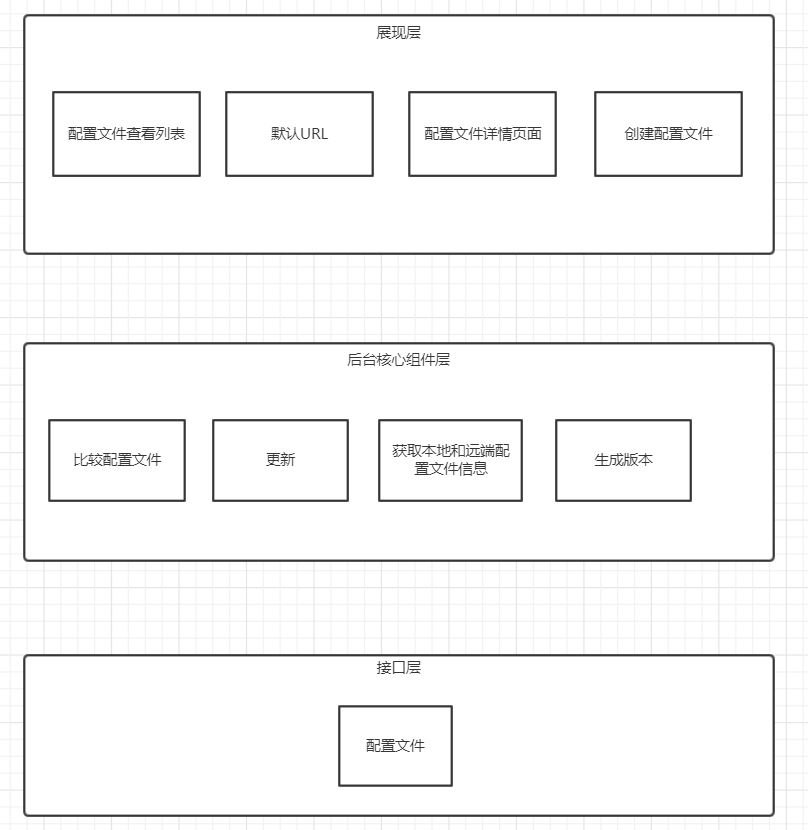


图1 总体逻辑框架图

## 4.1 系统运行环境

### 硬件平台：

主机为普通PC

* CPU：主频2.0G以上，核心数量2个（含）以上；
* 内存：4G（含）以上；
* 有效硬盘容量：200G（含）以上；
* 分辨率：推荐使用1920\*1080像素。

### 软件环境

* 操作系统： OS：Windows XP，Windows Vista、Windows 7、Windows10；
* 浏览器：IE，Chrome，Firefox。
* 开发语言：C#
* 开发工具：VS2015, Git

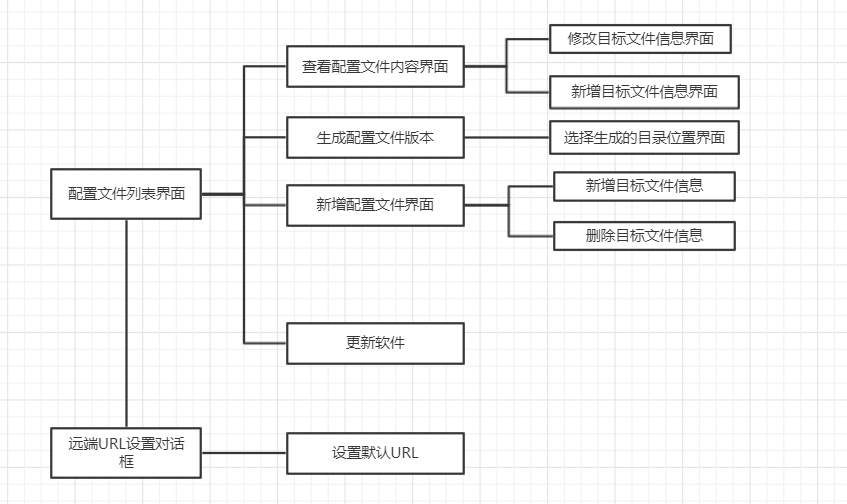
### 网络环境

* 支持有线网络和无线网络

# 5. 界面设计

软件自动版本更新系统主要实现比较配置文件信息、创建配置文件、查看修改配置文件、生成版本、自动下载更新等功能。主要界面设计如下：

* 配置信息列表主界面：显示程序本地和远端的配置文件条目在主界面，允许查看修改本地配置文件信息，允许使用远端配置文件进行更新，允许新增配置文件。当查看本地配置文件信息时进入查看配置文件信息界面；当新增配置文件时进入新增配置文件界面。
* 查看配置文件信息界面：显示某个配置文件中的内容，包括显示配置文件中目标文件的文件名、版本号、相应的哈希值以及目标文件的更新方式，允许对配置文件中目标文件进行增删改的操作。当修改配置文件中目标文件信息时进入修改目标文件信息界面；当点击新增时，进入新增目标文件信息界面。
* 新增配置文件界面：显示已添加的待更新的目标文件列表，列表包含目标文件的文件名、版本号、更新方式，当点击添加时，进入新增目标文件信息界面。
* 远端URL设置对话框：显示当前连接的远端资源的URL，可进行修改编辑。



# 6. 数据结构设计

## 逻辑结构设计

1. 配置文件表

存放各个配置文件，有配置文件名的属性

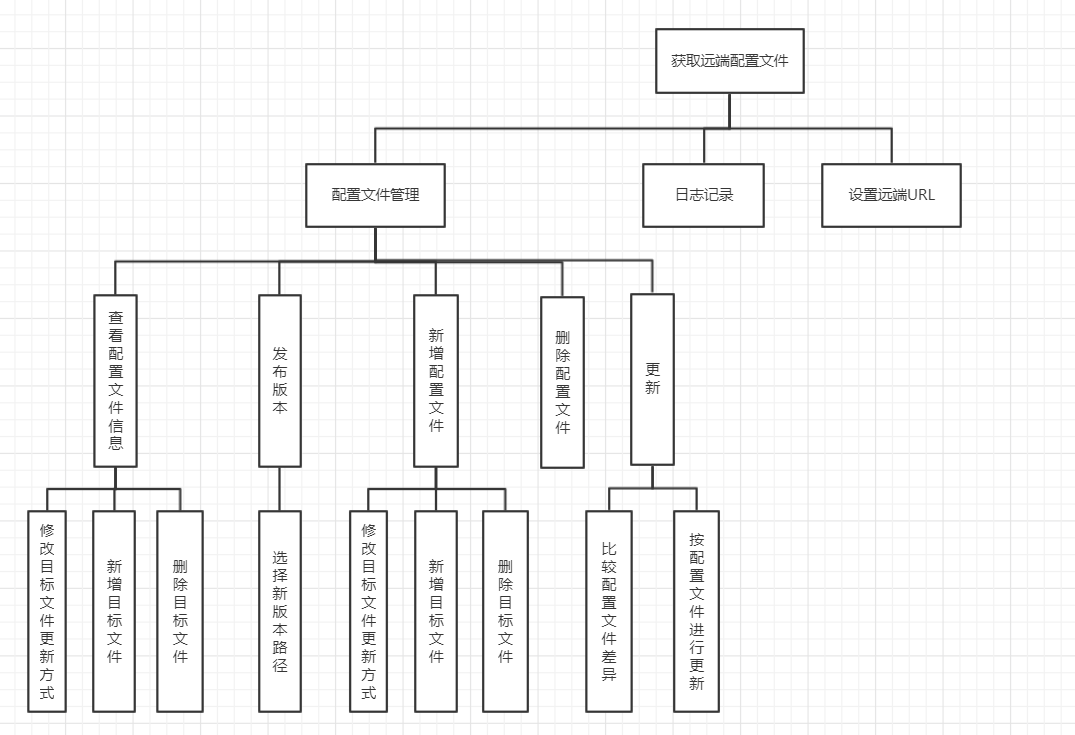
1. 配置文件
   1. 待更新目标文件名
   2. 待更新目标文件版本
   3. 待更新目标文件更新方式
   4. 待更新目标文件相应MD5值

## 物理结构设计

使用json文件保存相应的数据

# 7. 模块设计：

该系统启动后，先根据远端URL获取远端配置文件，然后进入配置文件管理模块。



## 7.1模块功能：

### 获取远端配置文件模块

程序启动后，先是访问预设好的URL访问远端资源，从远端URL获取远端的配置文件；

### 设置远端URL

通过修改远端URL来访问不同的远端资源，或为后续远端URL服务器变更做预留准备；

### 配置文件管理模块

该模块包括各个配置文件的增删改查，且允许使用各个配置文件进行生成版本，仅能使用远端配置文件进行更新；

### 查看配置文件信息模块

该模块显示配置文件中待更新的目标文件的信息，包括目标文件的文件名、版本号、更新方式、以及相应的MD5值；

### 修改目标文件更新方式

通过下拉框方式修改目标文件的更新方式；

### 新增目标文件

可以把本地某文件作为待更新的目标文件，添加至配置文件中；

### 删除目标文件

删除配置信息中的目标文件；

### 发布版本

通过点击发布版本，将配置文件和相应的目标文件拷贝复制到指定的目录下；

### 新增配置文件

软件允许创建一个新的配置文件，且可以往新的配置文件中添加新的待更新的目标文件，可进行增删改操作；

### 删除配置文件

删除一个已有的配置文件；

### 更新模块

将本地当前配置文件与获取的远端配置文件进行比较差异，按差异从远端获取相应的目标文件以进行远端配置文件中预设好的更新方式进行更新。

### 日志记录模块

使用log4net记录程序运行时的状态日志。日志的级别使用的DEBUG ， INFO ，WARN ，ERROR四种。

# 8.异常处理机制：

## 8.1异常检测机制：

该软件系统发生异常时主要是采用自动向上层汇报的机制。

## 8.2异常处理机制：

### 8.2.1 出错信息:

（1）添加待更新目标文件时，待目标更新文件不存在，抛出FileNotExistError异常；

（2）获取远端配置文件失败，抛出GetRemoteConfigError异常；

（3）打开配置文件失败，抛出FileOpenFailedError异常；

（4）更新软件失败，抛出UpdateError异常；

### 8.2.2 措施

（1）FileNotExistError： 弹出消息框提示用户，忽略本次添加操作

（2）GetRemoteConfigError：弹出消息框提示用户，并将更新按钮设置为不可用

（3）FileOpenFailedError：弹出消息框提示用户，忽略本次查看配置文件的操作

（4）UpdateError： 弹出消息框提示用户，忽略本次更新。

# 9. 日志记录规范：

## 9.1 日志库

Log4net

## 9.2 日志等级划分：

日志共分为五个等级，FATAL、ERROR 、WARN、INFO、DEBUG。

* DEBUG级别

用于记录调试时体统的运行状态，类似于调试过程中断点的作用。

* INFO级别

用于记录程序正确执行的信息。例如，添加一个新的配置文件到一个表里，可以先记录一个INFO日志：“添加XXX，id：XXX-XXX”。添加过程可以放到一个try-catch语句中，如果没有异常，可以在catch语句的后面在记录一个成功执行的INFO级别日志：“添加XXX，id：XXX-XXX成功”。应对最终用户具有实际意义。

* WARN级别

用于记录程序执行异常，但是又不影响程序正常运行的日志。例如某个操作要操作很多表，使用了实务操作。如果中间某个环节出现问题，事物回滚，那么就可以在catch中记录一个WARN级别的日志给予警告。

* ERROR级别

用于捕获程序的异常，一般记录在catch中。与WARN级别不同的是ERROR一般是一些影响业务数据正常工作的错误。例如某个数组的索引超出索引长度访问了，这时程序就应当记录一个ERROR日志。

* FATAL级别

用于捕获未知的异常，不在程序编码过程中使用，一般用在Global.asax文件中的Application\_Error事件中记录日志。专用于捕获那些编码中没有专门捕捉的异常信息。