

文章编号:1001-9081(2005)12Z-0429-02

批处理在软件配置管理中的应用

王敏,王攀

(中南民族大学电子信息工程学院,湖北武汉 430074)

(kkmnet@yeah.net)

摘要:从项目管理者角度,探讨了巧用软件配置管理工具 Microsoft Visual SourceSafe 提供的命令用户接口,用批处理方式实现高效率项目管理的方法。

关键词:项目管理;软件配置管理;版本标签;命令用户接口

中图分类号: TP311 **文献标识码:** A

软件配置管理(SCM)的重要性已越来越被项目管理者及软件开发人员所认可。许多优秀的软件配置管理工具也应运而生。Microsoft Visual SourceSafe(VSS)以其简单易用、方便高效、与 Windows 操作系统及微软开发工具高度集成等优点,在软件项目开发中得到广泛运用。

1 管理的重要性

软件工程项目与其他产品的工作项目一样要高质量高效率地完成和交付,不仅取决于所采用的技术,还取决于计划和管理的水平。所以现在的软件开发是把管理放在第一位。

1.1 软件配置管理的概念

软件生存期各个阶段的交付项,包括描述程序的文档、程序(源代码或可执行代码),以及包含在程序内部和外部的数据,它们组成了整体软件配置。软件配置管理就是对这些交付项修改的管理,它是在软件项目开始投入时开始,直至软件被废弃后才终止的一组跟踪和监控活动。

1.2 软件配置管理的重要性

在软件生产或用户使用过程中,修改是不可避免的,修改的起源在于:

- a) 新的商业或市场条件引起的产品需求或业务规则的变更;
- b) 新的用户需求,要求修改系统功能、数据或其他服务;
- c) 企业改组导致项目优先级或软件工程小组的结构变更;
- d) 预算或进度的限制导致系统或软件产品重定义。

由此可见,一个开发组可以产生出成百上千个独立的交付项,在开发和维护的各个步骤中,需要不断地再加工、修改和提高这些交付项,因此任何交付项都可能有很多不同的版本,而且某一项的任何一种版本都与其他交付项的特定版本有关。要防止相关交付项上、下、左、右不一致引起的错误以及意外的文件丢失,就必须进行配置管理。

2 VSS 的机能

软件配置管理是工作组软件开发中的重要方面,目的在于防止意外的文件丢失、允许反追踪到早期版本、并能对版本进行分支、合并和管理,可以随时方便地把握各种文件的修改状况,取得文件的各个版本。在软件开发和需要比较两种版本的文件或找回早期版本的文件时,源代码的控制是非常有用的。

VSS 是目前较常用的软件配置管理工具,它提供了完善

的配置管理功能,以及安全保护和跟踪检查功能。VSS 通过将有关项目文档存入数据库进行项目研发管理工作。用户可以根据需要随时快速有效地共享文件。文件一旦被添加进 VSS,它的每次改动都会被记录下来,用户可以恢复文件的早期版本,项目组的其他成员也可以看到有关文档的最新版本,并对它们进行修改,VSS 也同样会将新的改动记录下来。因此,用 VSS 来组织管理项目,使得项目组间的沟通与合作更简易而且直观。

VSS 可以同 VB、VC++、Visual J++、Visual InterDev、Visual FoxPro 开发环境以及 Microsoft Office 应用程序集成在一起,提供了方便易用、面向项目的版本控制功能。

VSS 在 Windows 应用软件开发及网络应用开发中均实用。VSS 可以处理由各种开发语言、创作工具或应用程序所创建的任何文件类型。它可以管理的文件种类有:文本文件、图像文件、二进制文件、声音文件、视屏文件。

3 VSS 的用户接口

VSS 提供的用户接口有三种,一个是通过 Visual Studio .NET 的集成开发环境(以后称 VS.NET)使用,另外也可用通过 VSS 独自の图形用户接口(后称 GUI)和命令用户接口(后称 CUI)使用。而通过 Visual Studio .NET 的集成开发环境及 VSS 的 GUI 对软件配置的管理均可用 VSS 的 CUI 来实现。对于编程者来说在 VS.NET 中使用比较方便,可以直接在开发窗口完成签出、签入操作;而对于项目管理者来说,利用 VSS 图形用户接口 GUI 进行管理更加直观,并能有效把握项目总体;而对那些处理量较大、处理时间较长以及经常反复使用的管理操作,利用 CUI 提供的各种命令行以批处理的方式来实现,更具优势。

4 用批处理实现软件配置管理

通常情况下,开发者都是利用集成开发环境 VS.NET 与 VSS 独自の GUI 进行软件配置管理,对于项目管理者来说,管理的范围往往不会具体到某一文件,而是整个项目的文件。此外,在项目开发周期中,需要进行产品的单体测试、结合测试、总体测试。每一次测试前,项目管理者都要取得新的版本再次建立解决方案,这样,管理者首先要启动 VS.NET,然后取得最新版本,在 VS.NET 环境下建立解决方案,这种同样的操作在开发过程中将重复多次进行,为了提高效率,在全项目的管理中将实现上述操作的命令行写入批处理文件,每次只需双击批处理文件,便可自动完成预定操作,管理效率比利用

收稿日期:2005-07-08;修订日期:2005-08-30

作者简介:王敏(1968-),女,讲师,主要研究方向:软件开发中的项目管理。

VS.NET 和 VSS 资源管理器 GUI 要高很多。既减轻了管理者的负担,又有效地避免了手动操作的失误。

这里以取得工程对应版本到建立解决方案探讨批处理文件的做成方法。

4.1 几个常用的批处理命令

a) 登录命令

同 VSS 的 GUI 一样,使用 VSS 命令行前,需要以用户名和密码登录,该处理的命令行如下:

```
set ssUser = [ username ]
set ssPwd = [ password ]
set ssDir = [ databasepath ]
```

其中:

username:用户名指定;

password:密码指定,鉴于安全性,这里可以不予指定,批处理执行时会提示输入密码;

databasepath: VSS 数据库路径指定;

登录处理完成后便可使用 VSS 命令行执行文件 SS.EXE,默认安装情况下,SS.EXE 在 C:\Program Files\Microsoft Visual Studio\VSS\win32 文件夹中。

b) 版本取得命令:

```
ss Get [ getpathname ] [ -G ] [ -H ] [ -I - ] [ -N ] [ -O ] [ -R ]
[ -V ] [ -W ] [ -Y ] [ -? ] [ buildpathname ]
```

命令行中:

ss: VSS 命令行工具执行文件名;

Get: 版本取得命令;

getpathname: VSS 数据库内的路径指定;

buildpathname: 取得版本的路径指定。

Get 命令参数选项说明如下:

-O: 重定向命令;

-V: 操作某项的指定版本;

-R: 表示取得某路径下的所有文件;

-G: 表示改变取得版本的保存路径;

-W: 指明本机副本是否只读(使用此参数,表示 writable);

-N: 在长短文件名两种模式间切换;

-I -: 任何环境下都不提示输入;

-Y: 指定用户名和密码;

-?, -H: 获得在线的帮助。

c) 解决方案建立命令

```
Devenv [ pathname ] / [ option1 ] [ option2 ]
```

命令行中:

Devenv : VS.NET 建立解决方案命令;

pathname: 建立解决方案的路径及文件名指定;

option1: 建立解决方案的方式指定;

option2: 建立解决方案的构成指定。

4.2 一个完整批处理程序

项目管理中类似以上的重复工作很多,均可将相关命令写入批处理文件来实现。这样,既能防止手动操作的不可靠性又能提高管理效率^[4]。

在取得版本,建立解决方案操作中,编写批处理文件时,除了上述相关的主要命令行之外,还需要对命令行方式下使用的交互手段进行一定控制,并且对执行过程中的错误也需要进行控制。下面是一个包含交互和错误处理控制的完整批处理程序,控制语句的功能均以注释方式在批处理文件中给出。

```
@ echo off
```

```
Echo 设置作业路径...
```

```
set workdir = C:\WORK\BlogX_1.1Beta
```

```
REM /s 如果要删除指定文件夹及其中包含的所有子文件夹和所有文件,参数设为/s。
```

```
REM /q rmdir 以 Quiet 模式执行。不给出提示信息,全部删除。
```

```
if exist %workdir% rmdir %workdir% /s /q
```

```
mkdir %workdir%
```

```
cd "%workdir%"
```

```
echo 解决方案对应版本取得
```

```
set path = %path% ; C:\Program Files\Microsoft Visual Studio\VSS\win32\
```

```
set ssUser = BuilderA
```

```
set ssPwd = teamdev
```

```
set ssDir = \\Win2003\VSS_BlogX\
```

```
REM ss.exe Get "%/BlogX" -W -R -I -Y -GL%workdir%
```

```
REM 取得指定版本编号的解决方案
```

```
ss Get "%/BlogX" -W -R -I -Y -GL%workdir% " -VL Version 1.1 Beta"
```

```
if errorlevel 100 goto VssBigErr
```

```
if errorlevel 1 goto VssSmallErr
```

```
echo 成功取得解决方案版本!
```

```
echo 解决方案建立中...
```

```
set path = %path% ; C:\Program Files\Microsoft Visual Studio .NET\Common7\IDE\
```

```
devenv %workdir%\BlogX.sln /rebuild Release
```

```
REM 由于没有 errorlevel 返回,判断 Release 文件是否做成
```

```
if not exist %workdir%\BlogCmd\bin\Release goto vsneterr
```

```
if not exist %workdir%\BlogXRuntime\bin\Release goto vsneterr
```

```
if not exist %workdir%\BlogXServiceProxy\bin\Release goto vsneterr
```

```
if not exist %workdir%\BlogXUploaderTest\bin\Release goto vsneterr
```

```
if not exist %workdir%\WeblogX\bin\Release goto vsneterr
```

```
if not exist %workdir%\WinBlogXControls\bin\Release goto vsneterr
```

```
echo 解决方案正常建立!
```

```
pause
```

```
exit 0
```

```
REM-----
```

```
:VssBigErr
```

```
echo ss.exe 错误编号 = 100。发生了严重问题。没找到数据文件夹,看是否签出了已经签出的文件。
```

```
echo 解决方案版本取得时发生了错误,处理终止!
```

```
pause
```

```
exit 100
```

```
:VssSmallErr
```

```
echo ss.exe 错误编号 = 1。轻度错误。从以下方面找问题:
```

```
echo ss Dir 已执行,但没有找到项目。
```

```
echo ss Status 已执行,但至少有一个项目是签出状态。
```

```
echo ss Diff 执行时,至少有一个文件是不同的。
```

```
echo 以上任何情况下,该命令虽然正常结束,但命令执行失败。
```

```
echo 解决方案版本取得时发生了错误,处理终止!
```

```
pause
```

```
exit 200
```

```
:vsneterr
```

```
Echo 建立解决方案时发生了错误,处理终止!
```

```
Pause
```

```
exit 300
```

4.3 批处理文件的执行

以上命令写入批处理文件后,以 .bat 后缀保存,这样在每次需要取得新版本或指定版本建立解决方案时,只需点击该批处理文件,便可简便完成原本复杂的操作。

能活用 VSS,选用适当的方法进行管理,将会给项目管理者带来很大的方便。

参考文献:

- [1] 杨文龙,古天龙. 软件工程[M]. 第2版. 北京: 电子工业出版社, 2004. 348 - 351.
- [2] 徐小青等. Visual Basic .NET 宝典[M]. 北京: 电子工业出版社, 2002.
- [3] ROCHE T. Essential SourceSafe[M]. Hentzenwerke Publishing, 2001.
- [4] 一色政彦. Visual Studio .NET によるチーム開発事始め [DB/OL]. http://www.atmarkit.co.jp/fdotnet/teamdev/teamdev04/teamdev04_02.html, 2003 - 11 - 05.