

软件配置管理工具综述

A Summary on Tools of Software Configuration Management

聂华北 (中国科技大学软件学院 安徽合肥 230026)

张艺超 (广州康大职业技术学院 广东广州 511363)

摘要: 阐述了软件研发行业中软件配置管理的适用性、必需性,并在此基础上介绍了软件配置管理的概念及其在软件研发中的地位 and 角色。结合行业当前实际使用的情况,从功能特点、优缺点、适应场景等方面综合对比了当前主流的一些配置管理工具。能有效帮助读者如何把配置管理策略应用到实践中。

关键词: 软件配置管理 版本控制 变更控制 配置审计 软件能力成熟度模型

1 引言

软件项目规模的日益增大,软件团队人员的增加,项目复杂性的不断加剧,软件版本不断变化以及多平台开发环境的采用,使得软件开发面临越来越多的问题,其中包括对当前多种产品的开发和维护、保证产品版本的精确(版本管理)、重建先前发布的产品、加强开发政策的统一和对特殊版本需求的处理等等,所有的这一切,必须采取各种管理措施和手段,而这些措施和手段也必须要依据和可度量的方法,这就是软件配置管理的核心内容。

2 软件配置管理介绍

软件配置管理,简称 SCM,关于它有多种定义。在 Wayne Baich 的《SCM Coordination for Team Productivity》一文中,对软件配置管理进行了定义:“协调软件开发使得混乱减到最小的技术叫做软件配置管理,它是一种标识、组织和控制修改的技术,目的是使错误达到最小并最有效地提高生产效率。”它贯穿整个软件生命周期并应用于整个软件工程过程。是软件工程中用来管理软件开发的规范,也是 CMM(软件能力成熟度模型)二级中的关键过程域。SCM 是软件质量改进的核心环节,它贯穿于整个软件生命周期,为软件改进提供了一套解决办法与活动原则。从 Anne Mette Jonassen Hass 的《Configuration Management Principles and Practice》一文中对 SCM 的理解“对系统生命周期内所选定的中间工作产品、产品组件以及产品的唯一标识、

受控存储、变更控制和状态报告。”中,可以很好地体现 SCM 的角色:它加强了软件的可靠性与质量,它提供了一个有效的软件开发与维护方案,用于识别和控制文档、代码、接口和数据库等软件元素。具有有效地控制复杂的软件版本、提供更好的并行开发方法、提供规范的软件测试和软件 BUG 以及修改流程、提供安全保密的备份措施、提供规范化的项目管理信息与软件产品信息等重要功能。

早期的手工作坊式的个人软件开发的局限性体现在:独立开发、手工控制共享内容、原始的备份方式和无序的开发流程等。例如,当在两个模块间共享的一个关于消息传递的定义需要扩充时,无法保证两个模块都在扩充后的规范下工作;在联试阶段,发现几个模块间关于一些传递数据的文件命名规范不同。现在的软件都是由团队甚至多团队合作开发完成的,有专门的研发小组来共同分析、设计、编码和维护,并由专门的测试小组对已完成编码调试的软件进行全面的测试。在软件开发这个庞大而复杂的过程中,需要涉及到各个方面的人员,信息的交流反馈不仅仅是在研发小组的成员之间及各个研发小组之间,还存在于客户和研发者之间。所有的这些交流反馈意见信息都有可能对软件的修改。在这个工程中,由于软件开发所固有的特征,可能会形成众多的软件版本,而且我们不能保证不出现错误的修改,由此而引发的主要问题有:如何对研发小组各成员所作的修改进行统一汇总;如何保留修改的轨迹,以便撤销错误的改动;如何

对在研发过程中形成的软件各个版本进行标识,管理及差异识别等等。

由以上对软件配置管理的定义以及现在软件项目管理的困难,可以确定,软件配置管理的目标是标识变更、控制变更、确保变更和报告变更,它主要完成以下几种任务:标识、版本管理、变更控制、配置审计和配置报告。

3 各软件配置管理工具的介绍

3.1 StarTeam

在真正大型的项目工程中,有越来越多的开发小组的数量,麻烦也随之而增加。并且合作者分散在不同的部门或不同的地方,此时项目组织中存在的噪音和混乱带来了沉重的经济负担!而单靠技术工具,或是下一个更快的编译器并不能解决面临的问题。即使已经使用了版本控制和软件配置管理工具和软件组织,但很快我们会发现生产力的提高比预期的要低,具体原因分析如下:

(1) 不同的项目组使用不同的工具,降低了效率。但搞统一是不合理的,违背自然规律;

(2) 不同工具之间有“隔阂”,难于重用数据,也减少了项目组之间的交流。

已有的版本控制工具不能支持远程用户,或者性能明显退化,如一个基于文件服务器的工具,当在 WAN 或 Internet 上使用时,会变得非常迟缓。StarTeam 能很好地解决以上出现的问题,它具有:

(1) 与 PVCS 和 SourceSafe 的互操作性;

(2) 完全的地理位置独立性;

(3) 客户/服务器体系结构,为 Internet 和 WAN 开发环境特别优化;

(4) 能够运行在任何 JAVA 平台上的统一的客户端应用;

(5) StarTeam 支持开发队伍建立一个协同工作的稳固基础的需要。

StarTeam 提供了真正的协同工作,使得开发环境内外的所有的功能范畴和团队能够积极参与共同的活动。是一个集合了版本控制、构建管理 (Build Management) 和缺陷跟踪系统为一体的软件,同时还具有以下额外的功能:

(1) 透明的跨引擎(不同的工具)互操作性;

(2) 一个提供访问你的所有软件资源的协同框架;

(3) 允许已有的用户继续使用其版本控制工具的能力,保护已有的投资;

(4) 完全的位置独立性;

(5) 平台独立性 (LAN/WAN/Internet/WEB);

(6) 虚拟的信息资源视图,允许每个项目成员将信息资源看作只与自己有关,并支持直观的进度、生命周期状态、产品版本、基线版本和私有配置。

这些额外的功能特性并不仅仅提供了对软件配置管理的支持。更重要的是,在软件配置管理的基础上建立技术协同。StarTeam 所提供的协同能力,能够真正提高生产力并取得竞争优势:所有成员主动合作和共享相关资源,实现更快、更好的目标。

3.2 PVCS (Version Manager)

PVCS 能够实现源代码、可执行文件、应用文件、图形文件和文档的版本管理;它能安全地支持软件并行开发,对多个软件版本的变更进行有效的控制管理。PVCS 不仅具备版本管理、建立管理、构造管理、问题追踪这些基本的功能模块,还融合了需求管理、需求变更管理技术,并支持工作流程,以至 Internet/Intranet 应用的异地通讯和管理功能。其主要的功能有:

(1) PVCS Version - Manager: 存储任何类型的文件的变更资料,可快速得到系统中任何文件的各个版本;

(2) PVCS Configuration - Builder: 为软件系统提供了可靠的自动重建过程。它保证系统在任何时候对某一发布的产品准确地进行重建,避免发生错误,同时自动地对修改过的模块重新编译以节省时间;

(3) PVCS Tracker: 在整个开发过程中确定和追踪软件的每一变更的要求;

(4) PVCS Notify: 将软件状态的变更通过 E-mail 通知组织机构中的其他成员;

(5) PVCS Reporter: 为 GUI 界面环境提供一个客户报表工具,使用它能很容易地生成和存储多个项目的报表;

(6) PVCS Production - Gateway: 提供了局域网与大型机 MVS 系统双向同步互联;

(7) PVCS Developer's Toolkit: 为 PVCS 客户提供

了应用程序开发接口 (API), 使项目信息通过编程访问;

(8) PVCS Requisite - Pro: 实现了先进的需求管理, 以及需求变更管理功能。

PVCS 能够支持所有硬件系统和操作系统平台的唯一产品, 能够有效地管理任何对象类型并且可以在独立的工作站上运行, 且能跨越单个 LAN 或多个 LAN。

3.3 ClearCase (CC)

CC 是 ROSE 构件的一部分, 主要应用于复杂的产品发放、分布式团队合作、并行的开发和维护任务。可以控制 word, excel, powerpoint, visio 等文件格式, 对于不认识的格式可以自己定义一种类型来标识。

ClearCase 用资源重用的方法帮助开发团队使他们的软件建立得更加可靠。解决了对当前多种产品的开发和维护, 保证产品版本的精确, 重建先前发布的产品, 加强开发政策的统一和对特殊版本需求的处理等等软件开发中的尖锐问题。CC 功能全面并且强大, 主要有:

(1) 版本控制

版本控制功能记录了对软件开发进程中一个文件或目录的多个版本历史信息, 可对所有文件系统对象 (包括源代码、二进制文件、文件、文档、测试包、目录以及连接等等) 进行版本的控制, 同时提供了先进的版本分支和归并功能用于支持并行开发, CC 版本树为并行开发分支提供了可视化历史记录。CC 支持所有符合 Windows 命名规范的文件类型;

(2) 工作空间管理

CC 通过使用 CC View 给每一个开发者提供了一致、灵活的可重用的工作空间域, 以保证开发人员拥有自己的开发环境或工作空间。同时 CC 提供了对版本进行透明化访问的功能通过 VOB 机制, CC 可以让开发者和应用者以一种标准文件目录树的形式访问 VOB。

(3) 建立管理

CC 拥有自己的建立程序 clearmake 和 omake, 并可兼容其他的 make 方法。C 自动产生软件系统构造文档信息清单, 而且可以完全、可靠的重建任何构造环境。也可以通过共享二进制文件和并发执行多个建立脚本的方式支持有效的软件构造。

(4) 过程控制。

CC 有一个灵活、强大的功能, 可以明确项目设计的流程。自动的常规日志可以监控软件被谁修改、修改了什么内容以及执行政策。

CC 提供一种监视和控制开发过程的机制, 支持质量标准的实施与保证, 如 ISO9001, CMM 等。主要应用于复杂的产品发放、分布式团队合作、并行的开发和维护任务, 包括支持当今流行软件开发环境 Client/Server 网络结构。对于它所支持的平台, CC 通过 TCP/IP 来连接客户端和服务器。另外, CC 拥有的浮动 License 可以跨越 UNIX 和 Windows NT 平台被共享。

3.4 Visual SourceSafe (VSS)

VSS 将所有的项目源文件以特有的方式存入数据库。开发组的成员不能对该数据库中的文件进行直接的修改, 而是由该版本管理器将该项目的源程序或是子项目的源程序拷贝到各个成员自己的工作目录下进行调试和修改, 然后将修改后的项目文件作 Check in 提交给 VSS, 由它进行综合更新。当某个成员向 VSS 中添加文件时, 该文件将会被备份到数据库中, 以便所有的成员都能共享该文件。每个成员对所有的项目文件所作的修改都将被记录到数据库中, 从而使得修改的恢复和撤销在任何时刻, 任何位置都成为可能。小组的成员可能得到该项目的最新版本, 对它进行修改, 并保存一个新的版本。其基本功能如下:

(1) 组内协调: 在一般情况下, 确保在任何时刻都只有一个成员对某个特定的文件进行修改, 这样可以防止文件被其他成员的修改意外更新;

(2) 版本跟踪: 对老版本的源代码和其他文件进行归档和跟踪, 而且这些版本能够被重新得到以便进行 bug 跟踪或其他目的;

(3) 跨平台开发: 支持同一代码在跨多个开发平台时的版本控制;

(4) 重用或面向对象代码: 跟踪哪些程序使用了哪些代码可被重用的模块;

(5) 归档和恢复: 在 VSS6.0 中这两个操作是在一个用户界面友好的 VSS 管理员 wizard 中进行的;

(6) 项目之间的差异比较: 允许用户在不同的项目之间进行差异比较;

(7) 历史信息过滤: VSS 6.0 支持查看那些没有标签的文件和项目的历史。

VSS 可以很容易地与 Microsoft Access、Visual Basic、Visual C++、Visual FoxPro 和其他的开发工具集成在一起,一旦 VSS 集成到开发环境中,就可以像控件一样使用,能够很好地体现出 VSS 的易用性和强大功能。

3.5 Concurrent Versions System (CVS)

CVS 是在原先 Unix 体系里很成熟的 SCCS 和 RCS (SCCS: Source Code Control System, RCS: Revision Control System) 的基础上开发出来的。几乎每一个 Linux 的发行版本里都能看到 CVS 系统。

CVS 的基本工作思路是在一台服务器上建立一个仓库,仓库里可以存放许多不同项目的源程序。由仓库管理员统一管理这些源程序。这样,就好象只有一个人在修改文件一样。避免了冲突。每个用户在使用仓库之前,首先要把仓库里的项目文件下载到本地。用户做的任何修改首先都是在本地进行,然后用 CVS 命令进行提交,由 CVS 仓库管理员统一修改。这样就可以做到跟踪文件变化,冲突控制等等。

CVS 的特点是它的客户机/服务器存取方法使得开发者可以从任何因特网的接入点存取最新的代码;无限制的版本管理检出 (check out) 的模式避免了通常的因为排它检出模式而引起的人工冲突;可用于各种平台,包括 Linux、Unix 和 Windows NT/2000/XP 等等。CVS 的主要功能有:

(1) 版本管理,保存所有代码、文件更改的历史记录。对代码进行集中统一管理,可以方便查看新增或删除的文件,能够跟踪所有代码改动痕迹。可以随意恢复到以前任意一个历史版本;

(2) 完善的冲突解决方案,可以方便的解决文件冲突问题;

(3) 代码权限的管理。可以为不同的用户设置不同的权限。可以设置访问用户的密码、只读、修改等权限,而且通过 CVS ROOT 目录下的脚本,提供了相应功能扩充的接口;

(4) 支持方便的版本发布和分支功能;

(5) 代码集中的配置。CVS 内建了客户机/服务器存取方法,所以任何一个可以连到因特网上的开发者都可以存取在一台 CVS 服务器上的文件;

(6) 调整代码。与传统的版本控制软件不同,当多个开发者对同一个文件作了修改,CVS 会检测,并且自动合并那些改变——CVS 会很小心,并且将会自动合并那些只要不是对代码的同一行所作的改动。

4 总结

软件配置管理是软件工程中用来管理软件开发的规范,也是 CMM (软件能力成熟度模型) 二级中的关键过程域。SCM 加强了软件的可靠性与质量,它提供了一个有效的软件开发与维护方案,用于识别和控制文档、代码、接口和数据库等软件元素。项目的开发中,我们除了要有好的工具外,还要制定一系列的软件变更控制及版本控制制度 (SCM 的核心),并且很好地实行。做好软件配置管理有利于对整个开发团队进行管理,软件本身是思维逻辑和数据的固化,而思维逻辑和数据是无形的,软件配置管理就是对这些无形的思维逻辑和数据进行变更记录,使之可以度量、统计分析和

参考文献

- 1 蔡明军. 迈好软件开发规范管理第一步: 软件配置管理. 软件世界—赛迪网, 2003.
- 2 胡杰. 版本管理器 Visual SourceSafe 6.0 实用指南. 2003.
- 3 蔡伟丽. 软件配置管理方法与工具. 中国电子科技工业集团公司第 54 所.
- 4 向为, 伍微, 王浩. 软件配置管理解决方案. 国防科技大学电子科学与工程学院卫星导航定位研发中心, 2004.
- 5 郑人杰. 论软件开发管理的作用—从 PVCS 谈起. 清华大学软件技术中心, 2002.
- 6 Anne Mette Jonassen Hass. Configuration Management Principles and Practice. In: Addison Wesley, a Pearson Education Company, 2003. 245 - 247.
- 7 张萍, 易东云. 软件配置信息设计实现方法. 计算机系统应用, 2003, 12(2): 60 - 62.