

基于基线的软件配置管理版本控制

任永昌, 朱 萍, 李仲秋

(渤海大学 信息科学与技术学院, 辽宁 锦州 121013)

摘 要: 版本控制是软件配置管理的重要功能, 负责为配置库中的所有元素自动分配版本标识, 并保证版本命名的唯一性。文中运用软件工程的原理并结合版本控制的实践, 首先指出为了有效地进行版本控制而引入基线管理, 并通过图形表示最常用的软件项目基线; 然后研究版本管理, 包括版本管理的功能、版本树、版本的串行与并行演变过程, 并通过图形给出了版本演变历史和组件版本的变化过程; 最后研究版本控制的主要内容, 包括检入检出控制、分支和合并、历史记录。结果表明, 在配置管理过程中通过基线进行版本控制, 是保护代码资源、提高开发效率、加强软件项目管理的有效途径。

关键词: 基线; 软件配置管理; 版本控制; 检入检出控制

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X (2012)11-0113-03

Version Control Study of Software Configuration Management Based on Baseline

REN Yong-chang, ZHU Ping, LI Zhong-qiu

(College of Information Science and Technology, Bohai University, Jinzhou 121013, China)

Abstract: Version control is the important function of the software configuration management. It's responsible for all the elements in the configuration repository assigns a version identifier automatically, and ensures the version named only. Applying the principle of software engineering and combining the practice of version control, first point out that introduce baseline management in order to control version efficiently, and with graph represent the most commonly used software project baseline. Second, study version management, including version management features, version tree and the serial and parallel evolution of the version, then give the change process about the history of version evolution and the component version through the graphical. Finally, study the main context of the version control, including check in-check out control, branching and merging, historical records. The results show that with baseline to do version control in the configuration management process is the effective way to protect the code resources, improve the development efficiency and strengthen the software project management.

Key words: base line; software configuration management; version control; check in-check out control

0 引言

软件配置管理是规范、高效的软件开发基础结构, 是在项目开发过程中, 标识、控制和管理软件变更的一种管理活动。随着软件团队人员的增加, 软件版本不断变化, 由于开发时间的紧迫以及多平台开发环境的采用, 出现了诸如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、保持开发规范的一致性、对特殊版本要求的处理等一系列问题, 解决这些问题的重要途径之一就是软件配置管理^[1]。

有效的配置管理, 这使得用户期望软件系统能够很好地适应外界要求的变化, 使日后的维护和升级得到保证^[2,3]。版本控制是软件配置管理的重要活动, 因为软件变更常常带来软件版本的改变与新版本的发布, 必须进行有效的控制。版本控制的对象包括文档、源代码、规范、可执行文件等, 目的在于对软件开发过程中的这些对象提供有效的跟踪手段, 便于对版本进行检索和跟踪, 避免文件的丢失和覆盖^[4]。

1 基 线

基线是软件开发过程中的关键里程碑, 当某一个软件配置项纳入基线管理后, 就是后续工作的标准, 只有经过管理人员授权后才能改变这个标准。基线又可细分为如下3类:

(1) 功能基线。是最初批准的功能配置标识, 可以是甲乙双方签订的有关软件实现功能的协议书, 也

收稿日期: 2012-02-27; 修回日期: 2012-05-28

基金项目: 国家自然科学基金项目(70871067); 2011年辽宁省东欧及独联体国家重点引智项目; 2011辽宁省科学事业公益基金

作者简介: 任永昌(1969-), 男, 教授, 博士, 从事信息处理、软件工程、软件项目管理等研究; 朱 萍(1987-), 女, 硕士研究生, 从事软件项目管理研究。

可以是需要分析结束后确定软件系统功能,或者是上级主管部门下达的软件规格说明。功能基线确定软件需要实现的功能。

(2)指派基线。是最初批准的指派配置标识,通常是需求分析阶段工作结束时,经过正式评审和批准后的软件需求规格说明。通过指派基线,可以将工作分派给相关的组织或个人。

(3)产品基线。是最初批准的产品配置标识。通常是系统组装完成并经过系统测试,软件能与系统其他部件正确连接,达到规定的功能和性能,通过正式评审,可以将产品投入使用。

一般最常用的软件项目基线如图 1 所示。基线提供了软件生命周期中各个开发阶段的一个特定点,使得连续的开发工作通过这些特定点分割开来,便于对阶段性成果进行检查。处于基线管理下的版本一般不是同一时间产生的,但具有在开发的某一特定步骤上相互一致的性质^[5,6]。基线作为检查点,正式发行的产品必须是经过控制的基线产品。

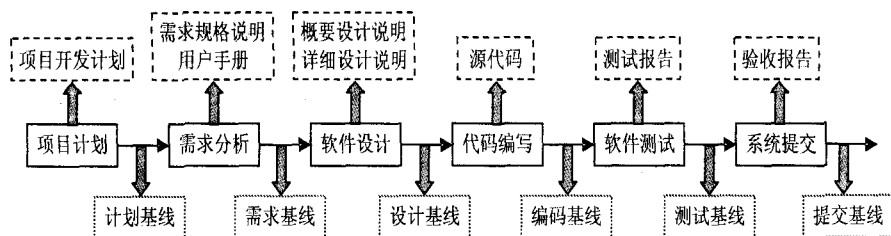


图 1 软件项目基线

2 版本管理

版本管理是软件配置管理的基础,通过版本管理保护开发的软件资源。版本管理的作用如图 2 所示。软件系统版本随着软件开发工作的进展而演变,从而构成复杂的版本空间。版本和基线的关系可以描述为:配置管理保存已完成的版本、创建新版本、合并多个版本,并提供一种策略来选择版本创建基线^[7,8]。

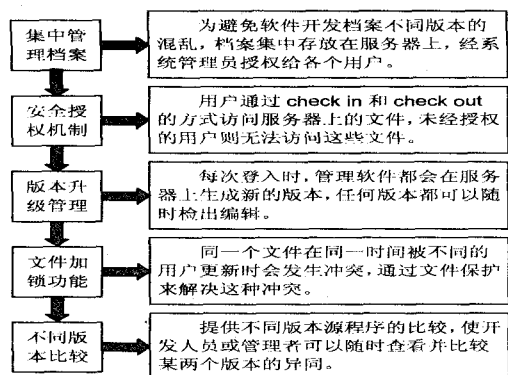


图 2 版本管理的功能

一个系统版本就是一个系统实例,在某种程度上

有别于其它系统实例。系统新版本可能有不同的功能、性能,可能修改了系统错误。有些版本可能在功能上没有什么不同,只是为不同的硬件或软件配置而设计的。如果版本之间只有细微区别,有时就把其中的一个版本称作另一个版本的变体。

一个系统的发布版本是要分发给客户的版本。每个系统发布版本都应该包含新的功能或是针对不同的硬件平台^[9]。一个系统的版本要比发布版本多得多,因为机构的内部版本是为内部开发或测试而创建的,有些根本不会发布到客户手中。

版本的演变一般有两种方式:串行演变和并行演变。串行演变是由当前的最新版本形成新版本,并行演变是由一个最新版本形成多个新版本,两种演变方式也经常交织在一起进行,演变的结果称为版本树^[4]。版本树示例如图 3 所示。

在图 3 中,共有三条版本链,分别是“1.0-1.1-1.2-1.3-1.4-1.5”、“1.0-1.1-1.1.1-1.1.2”和“1.0-1.1-1.2-2.0-2.1-2.2-3.0-3.0.1”。“1.0-1.1-1.2-1.3-1.4-1.5”是串行演变。

“1.1-1.2”和“1.1-1.1.1”是并行演变。

为了建立特定的系统版本,必须确定系统组件的版本。在一个大型软件系统内,有数以百计的软件组件,其中每种组件都可能有多不同的版本。版本管理规程应该规定明确地标识每个组件版本的方法,这样在需要进一步变更时,可以查找到组件的具体版本。组件版本的变化过程如图 4 所示。

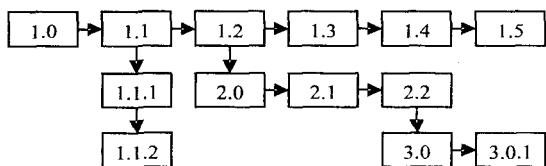


图 3 版本树示例

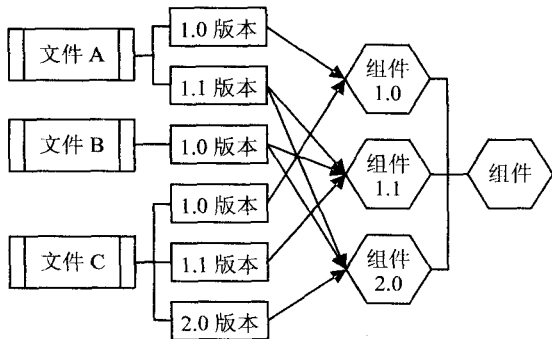


图 4 组件版本的变化过程

软件配置管理的过程,还应该包括对产品组件的

质量控制。在图 4 中^[10],组件 1.0 版本与 1.1 版本的组成成分不同,1.0 版本由 A、C 两个文件组成,而 1.1 版本由 A、B、C 三个文件组成。组件 1.1 版本与组件 2.0 版本都由 A、B、C 三个文件组成,但文件的版本不同,组件 1.1 版本由文件 A 的 1.1 版本、文件 B 的 1.0 版本和文件 C 的 1.1 版本构成,而组件 2.0 版本由文件 A 的 1.1 版本、文件 B 的 1.0 版本和文件 C 的 2.0 版本构成。

3 版本控制

版本控制是软件配置管理的核心功能。版本命名系统保证所有置于配置库中的元素命名的唯一性^[11]。软件配置管理版本控制之所以在基线的控制之下,是因为处于基线控制的软件配置项版本,表明已完成了一定的阶段性工作,版本处于被锁定的状态;而非基线控制的版本,软件开发人员可以随意修改而不必经过审批流程。被锁定的版本如果需要修改,版本控制的基本流程如图 5 所示。

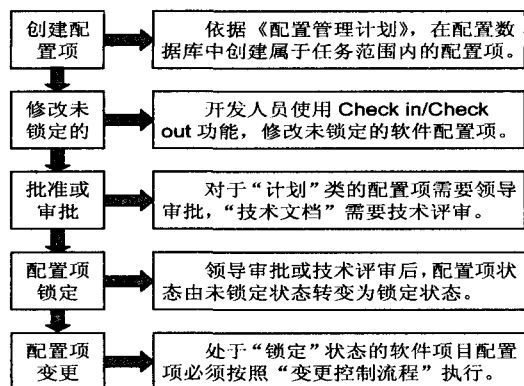


图 5 版本控制的基本流程

现在的版本控制通常由 CASE 工具来支持。工具用于管理对每个系统版本的存储,并控制对系统组件的访问。这些组件必须能够从系统中抽取出来进行编辑,当将其重新放入系统的时候,就构成了一个新的系统版本,由版本管理系统给它一个新的名字。版本控制包括:检入检出控制(Checkin/Checkout)、分支和合并(Branch/Merge)、历史记录(History record)。

3.1 检入检出控制

建立基线之后,配置项被保存在配置数据库中。这些配置项不能随意进行修改,但在很多时候仍然需要修改,修改后仍需要保存在配置数据库中,这就需要检入检出控制。“检入”就是将软件配置项从开发人员的工作空间存储到配置数据库的过程,“检出”就是将软件配置项从配置数据库取出并存储到开发人员工作空间的过程。检入检出控制过程如图 6 所示^[12]。

检入检出控制保证了软件工程师对访问对象的权

限,检出的同时锁定了这个对象,使得当前检出的版本在没有被转换前不能再更新这个变更的配置对象。

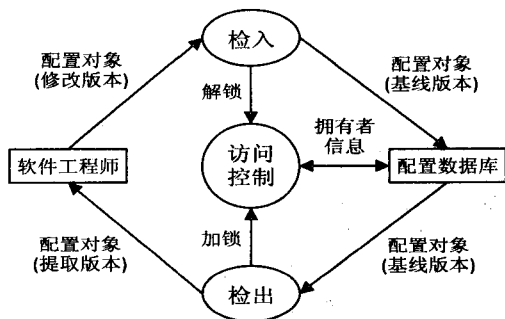


图 6 检入检出的控制过程

3.2 分支和合并

版本分支的人工方法是从主版本拷贝一份并做上标记^[13],版本合并的人工方法是将一个版本的内容拷贝到另一个版本上形成新版本或将两个版本的内容合并为一个新版本。

对文件来说,分支与合并的结果就是形成具有图结构的版本历史,即版本图,如图 7 所示^[6]。

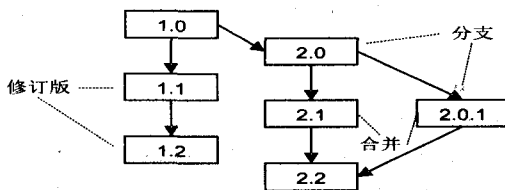


图 7 版本的分支与合并

3.3 历史记录

版本控制的历史记录,跟踪并记录了整个软件的开发过程,便于对在软件开发过程不同阶段新生的成果(包括源代码、文档资料等)进行对比分析,对已修改的软件配置项恢复到未修改新的状态,对不同人员所做的修改进行监视、监督和汇总,提高软件项目的管理水平的控制能力,同时历史记录记载了版本的修改时间、修改人员、修改描述等信息,便于分清责任并进行绩效考核管理。

4 结束语

版本控制逐渐成为软件配置管理中重要的控制过程,是软件项目管理领域重要的研究方向。配置管理的版本控制可以解决很多问题,包括根据不同用户的需求,配置不同的软件系统;保存系统老版本,为撤销修改或追溯问题提供支持;支持多个软件开发人员共同开发一个项目,或同时修改一个软件配置项;存储软件的多个版本,清晰地表明软件版本的演进过程。IEEE 对于基线的定义表明,版本、基线和变更虽然是不同的三个概念,但在软件开发过程又是紧密相连的,

(下转第 119 页)

(2)用改进的二维 Otsu 算法确定整幅图像阈值;

(3)边缘检测:采用 Candy 算子对图像进行检测从而确定比较准确的、符合实际的图像边缘信息;

(4)如果像素点为边缘点则对其进行 Bernsen 二值化处理,非边缘点则运用改进的二维 Otsu 算法对其进行二值化。

2.2 实验结果

为验证文中所提方法做了一系列的实验,图 d 给出了对图 a 图像按照文中所提方法进行二值化结果。作为对比,图 b 给出了局部阈值 Bernsen 法对同一幅图像进行二值化所得结果。图 c 给出了全局阈值改进的大律法对同一幅图像进行二值化得到的结果。

3 结束语

根据实验结果可以得出,文中提出的方法对于光照不均匀且对比度低的图像进行二值化的效果远远大于大律法和局部阈值法。虽然二维大律法增加了一些计算量,但通过改进其划分区域及加入松弛变量后计算量较传统的二维大律法有所减少。这种方法能很好地避免局部阈值法所产生的块效应和过多虚假信息,又比全局二值化能得到更好的二值化图像,并且又能保留图像的细节信息。

参考文献:

- [1] Otsu. A threshold selection method from gray-level histogram [J]. IEEE Trans on SMC, 1979, 9(1): 62-69.
- [2] 刘建庄,栗文青. 灰度图像的二维 Otsu 自动阈值分割法

(上接第 115 页)

因为软件变更涉及到版本的变化,变更又受到基线的控制^[13]。基于基线的版本控制,能确保软件配置项在软件生命周期中始终处于受控状态,并恢复到任一个历史状态,方便地进行软件项目的开发和管理。

参考文献:

- [1] 郭宁,周晓化. 软件项目管理[M]. 北京:清华大学出版社,2007.
- [2] 李玉龙,李长云. 软件动态演化技术[J]. 计算机技术与发展,2008,18(9): 83-86.
- [3] 王珍英. 配置管理在软件项目管理中的应用[J]. 计算机系统应用,2008,17(6): 101-104.
- [4] Seawhio P, Suwannasart T. A SCM workflow model for CMM organizations[J]. Software Engineering, 2003, 21(10): 253-260.
- [5] 韩万江,姜立新. 软件项目管理案例教程[M]. 北京:机械工业出版社,2006.
- [6] Ren Y C, Xing T, Quan Q, et al. Study on software configura-

[J]. 自动化学报,1993,19(1):101-105.

- [3] 郝颖明,朱枫. 二维 Otsu 自适应阈值的快速算法[J]. 中国图象图形学报,2005,10(4): 484-488.
- [4] Blayvas I, Bruckstein A, Kimmel R. Efficient Computation of Adaptive Threshold Surfaces for Image Binarization[C]//Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. [s. l.]: [s. n.], 2001.
- [5] 肖艳炜,张云. 改进的二维 Otsu 自动分割算法及其应用研究[J]. 计算机工程与应用,2007,43(7): 243-245.
- [6] Canny J. A Computational to Edge Detection[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1986, 8(6): 679-698.
- [7] Lee K K, Chan W K, Chen Q R. Chin Contour Estimation Using Modified Canny Edge Detector[C]//Proceedings of the 7th International Conference on Control, Automation, Robotics and Vision(ICARCV). [s. l.]: [s. n.], 2002: 770-775.
- [8] 杨晖,曲秀杰. 图像分割方法综述[J]. 电脑开发与应用, 2005(3): 21-23.
- [9] 张庆英,岳卫宏,肖维红,等. 基于边界特征的图像二值化方法应用研究[J]. 武汉理工大学学报,2005,27(2): 55-57.
- [10] 张楠,吴志斌,陈淑珍,等. 一种新的自适应二值化方法[J]. 计算机工程,2002,28(5): 184-186.
- [11] 陈强,朱立新,夏德深. 结合 Candy 算子的图像二值化[J]. 计算机辅助设计与图形学学报,2005,17(6): 1302-1306.
- [12] 王涛,徐娅萍,亢海龙,等. 结合 Candy 算子的图像二值化方法[J]. 微型电脑应用,2010,26(2): 4-7.
- [13] tion management of version control [C]//3rd International Conference on Information Management Innovation Management and Industrial Engineering. Kunming: [s. n.], 2010.
- [7] Westfechtel B, Munch B P, Conradi R. A layered architecture for uniform version management [J]. IEEE Transactions on Software Engineering, 2001, 27(12): 121-129.
- [8] 倪晓峰,赵文耘,张捷. 构件软件配置管理以及其版本控制技术[J]. 计算机工程与应用,2005,42(2): 94-96.
- [9] 袁肃蓉,王萍,黄万民,等. 基于 ClearCase 的软件配置管理环境的规划和实施[J]. 海南大学学报,2009,27(1): 75-79.
- [10] 任永昌,邢涛,鄂旭. 软件项目开发过程管理[M]. 北京:北京交通大学出版社,2010.
- [11] 王如龙,邓子云,罗铁清. IT 项目管理-从理论到实践[M]. 北京:清华大学出版社,2007.
- [12] 朱少民. 软件质量保证和管理[M]. 北京:清华大学出版社,2007.
- [13] 崔方圆. 支持分布式协同开发的软件配置管理系统研究[D]. 大连:大连海事大学,2009.