软件配置管理定义

软件配置管理是指在开发过程中各阶段,管理计算机程序演变的学科,它作为软件工程的关键元素,已经成为软件开发和维护的重要组成部分,它提供了结构化的,有序化的,产品化的管理软件工程的方法。它涵盖了软件生命周期的所有领域并影响所有数据和过程。ieee标准729-1983就配置管理的内容进行了规范的定义

标识 识别产品的结构、产品的构件及其类型,为其分配唯一的标识符,并以某种形式提供对它们的存取。

控制 通过建立产品基线,控制软件产品的发布和在整个软件生命周期中对软件产品的修改。例如,它将解决哪些修改会在该产品的最新版本中实现问题。

状态统计 记录并报告构件和修改请求的状态,并收集关于产品构件的重要统计信息。例如,它将解决修改这个错误会影响多少个文件的问题。

审计和审查 确认产品的完整性并维护构件间的一致性,即确保产品是一个严格定义的构件集合。例如,它将解决目前发布的产品所用的文件的版本是否正确的问题。

生产 对产品的生产进行优化管理。它将解决最新发布的产品应由哪些版本的文件和工具来生成的问题。

过程管理 确保软件组织的规程、方针和软件周期得以正确贯彻执行。它将解决要交付给用户的产品是否经过测试和质量检查的问题。

小组协作 控制开发统一产品的多个开发人员之间的协作。例如,它将解决是否所有本地程序员所做的修改都已被加入到新版本的产品中的问题。

软件配置管理是一套规范、高效的软件开发基础结构。作为管理软件开发过程有效的方法,早已被发达国家软件产业的发展和实践所证明。可以系统地管理软件系统中的多重版本全面记载系统开发的历史过程包括为什么修改,谁作了修改,修改了什么管理和追踪开发过程中危害软件质量以及影响开发周期的缺陷和变化。

下面简要介绍几个软件配置管理的基本概念,主要包括软件配置与配置项、基线、软件版本。

软件配置与配置项

软件配置是指一个软件产品在软件生存周期各个阶段所产生的各种形式机器可读或人工可读和各种版本的文档、程序及其数据的集合。该集合中的每一个元素称为该软件产品软件配置中的一个配置项 。

软件配置项即软件配置管理的对象。软件开发过程中,产生的所有信息构成软件配置,它们是代码源代码、目标代码以及数据结构内部数据、外部数据、文档技术文档、管理文档、需求文档、报告,其中每一项称为软件配置项,软件配置项是配置管理的基本单位。同时,开发过程中使用的环境,如操作系统、各种支撑软件、工具等,也可纳入软件配置管理范围。图为常规状态下软件开发阶段的配置项定义。

基线和配置管理数据库在软件开发过程中,由于各种原因,可能需要变动需求、预算、进度和设计方案等,尽管这些变动请求中绝大部分是合理的,但在不同的时机作不同的变动,难易程度和造成影响差别甚大,为了有效地控制变动,软件配置管理引入基线的概念。

基线标志软件开发过程的各个里程碑,.任一个sci,例如,设计师说明书,一旦形成文档并复审通过,即形成一个基线,它标志开发过程中一个阶段的结束。对于已成为基线的,虽然可以修改,但必须按照一个特殊的、正式的过程进行评估,确认每一处修改。相反,对于未成为基线的,可以进行非正式修改。最常见的基线如图所示'。

某个一旦成为基线,随即被放入项目数据库。此后,若开发小组中某位成员想要改动,首先要将它拷贝到私有工作区并在项目数据库中锁住,不允许他人使用。在私有工作区中完成修改控制过程并复审通过之后,再把修改后的推出并回到项目数据库,同时解锁。

建立基线的主要原因是重现能力、可追踪性和报告能力。重现能力是指返回并重新生成软件系统给定发布版本的能力。可追踪性是建立项目各种类型工件需求、设计、实现、测试等之间的横行依赖关系,其目的在于确保设计满足需求、代码实施设计以及使用正确代码编译生成可执行文件。报告能力来源于一个基线内容同另一个基线内容的比较,基线比较有助于程序调试并生成发布说明。

不可变的源对象经质量检查合格后所形成的新的相对稳定的配置成为软件版本。版本记录了配置项的演化过程。通常说的版本指的是已经交付给用户使用的产品的版本。在交付给用户使用以前,同样也会产生各种各样的版本。软件配置项在不同的时期,由于不同的要求会出现不同的组合,为了表明不同的特性,也会标明不同的版本。因此,每个软件配置项有一个版本组。他们彼此之间有特定的关系,这种关系用以描述其演变情况,通常软件配置项的版本组成树形结构。

在软件配置管理中,必须考虑如下一个问题,怎样的配置管理活动可以解决诸如采用何种方式标识和管理已存在程序的各种版本、在软件交付用户前后如何控制变更、利用什么办法来估计变更引起的其他问题等等类似问题,这些问题可归结到软件配置管理中的活动上面。配置管理主要包括十个活动访问和查找软件、在开发生命周期中实现各种变更、在开发生命周期中迁移变更、管理编译和构建过程、管理变更的分布、获得批准和签署、管理软件变更请求、协调组间通信、获得项目状态、追踪错误和定位等。而对于这十项活动所完成的功能进行

分析研究,可总结软件配置管理的主要功能流程示意如图所示。根据图,对于流程各项活动的归纳分析,软件配置管理流程中的活动可归划到配置标识、版本控制、变更控制、状态报告和配置审计五大功能当中去,我们称这五部分为软件配置管理的关键活动。

配置标识—在系统演化过程中表示其中的中间软件产品。

版本控制—记录每个配置项的发展历史,并控制基线的生成。

变更管理—包括在整个生命周期中控制中间软件产品的变化。

状态报告—记录和报告软件的变化过程。

配置审计—用于保证软件产品是依照需求、标准和合同开发出来的。

配置标识

配置标识是软件生存周期中产生的所有文档的总称。在软件开发过程中,随着软件生存期的向前推进,软件配置项不断增多。为了方便对新增的,以及对原进行变更而形成的新的版本进行管理和控制,使之管理方便且有序,因此而为其命名。

配置标识定义配置项的名称、与其他配置项的联系、版本信息等。在对软件配置项进行标识时,不但要考虑每个软件配置对象固有的名字、描述、资源列表和实际存在四个部分,还需注意软件配置项之间的关系以及变更控制方面的内容。

配置标识的主要活动由选择配置项、制定配置项标识方案以及存取方案组成,有效的配置标识是其他配置管理活动的前提。配置标识包括识别软件系统的结构、唯一表示独立的组建并使其可以某种方式存取。配置标识的目的是获得在软件整个生命周期中识别其组件的能力,配置标识告诉我们系统中的配置是什么,文件的版本是什么。

综上,配置标识是配置管理的基础,是唯一地标识软件配置项和各种文档,使它们可用某种方式访问。配置标识的目标是在整个系统生命周期中标识系统的组件,提供软件和软件相关产品之间的追踪能力。

版本控制

版本控制是软件配置管理中不可缺少的功能。所有置于配置库中的元素都应自动予以版本的标识,并保证版本命名的唯一性。版本在生成过程中,自动依照设定的使用模型自动分支、演进。除了系统自动记录的版本信息以外,为了配合软件开发流程的各个阶段,我们还需要定义、收集一些元数据来记录版本的辅助信息和规范开发流程,并为今后对软件过程的度量做好准备。当然如果选用的工具支持的话,这些辅助数据将能直接统计过程数据,从而方便我们软件过程改进,活动的进行。对于配置库中的各个基线控制项,应该根据其基线的位置和状态来设置相应的访问权限。一般来说,对于基线版本之前的各个版本都应处于被锁定的状态,如需要对它们进行变更,则应按照变更控制的流程来进行操作。

变更管理是指在软件开发的整个周期中对软件的变化进行控制和追踪。从对于基线的定义中我们可以发现,基线是和变更控制紧密相连的。也就是说在对各个做出了识别,并且利用工具对它们进行了版本管理之后,如何保证它们在复杂多变的开发过程中真正的处于受控的状态,并在任何情况下都能迅速的恢复到任一个历史状态就成为了软件配置管理的另一重要任务。因此,变更控制就是通过结合人的规程和自动化工具,以提供一个变化控制的机制。

变更控制是软件配置管理的核心,它通过创建产品基线,在产品的整个生存周期中控制它的发布和变更,是在整个软件生命周期中控制对软件的变化。变更控制记录每次变化的相关信息。查看这些记录信息,有助于追踪出现的各种问题。记录正在执行的变化信息,有助于做出正确的管理决策。

软件的变更通常有两种不同的类型`功能变更和错误修改变更。功能变更是为了增加或者删除某些功能、或者为了改变完成某个功能的方法而需要的变更。这类变更必须经过某种正式的变更的代价比较小且对软件系统其他部分没有影响,或者影响很小,通常应批准这个变更。反之,如果变更的代价比较高,或者影响比较大,则必须权衡利弊,以决定是否进行这种变更。如果同意这种变更,需要进一步确定由谁来支付变更所需要的费用。如果是用户要求的变更,则用户应支付这笔费用否则,必须完成某种成本效益分析,以确定是否值得做这种变更。因此,功能变更涉及到软件额外费用的问题。

错误修补变更是为了修复漏洞而要进行的变更。它是必须进行的,通常不需要从管理角度对这类变更进行审查和批准。但是,如果发现错误的阶段在造成错误的阶段的后面,例如,在实现阶段发现了设计错误,则必须遵照标准的变更控制过程,把这个变更正式记入文档,把所有受这个变更影响的文档都作相应的修改。

在软件开发过程中变更是不可避免的,但是,变更的内容、变更的幅度都会直接影响到各项目,我们时时刻刻需要考虑到变更的波及面。例如,在一个瀑布模型的生命周期中,如果需求发生变化后,软件生命周期中各个阶段都有可能受到影响,这些阶段的所有相关的软件配置项都或多或少的受到影响。受到影响的各部位必须进行调整才能够应对这些变化所带来的冲击。

配置审计

软件的完整性实质是指开发后期软件产品能正确满足用户所提出的对软件的要求。而软件配置审计的目的就是要证实整个软件生存期中各项产品在技术上和管理上的完整性,同时,还要确保所有文档的内容变动不超出当初确定的软件要求范围,使得软件配置具有良好的可跟踪性,这是软件变更控制人员掌握配置情况,进行审批的依据'。

软件配置审计一般使用两种方法来确定变更得正确完成,即正式技术评审和软件配置审计。正式的技术评审在软件交付用户前实施。着重检查已完成修改的软件配置对象的技术正确性,评审者评价软件配置项,决定它与其它软件配置项的一致性。一般对所有的变更都要进行正式技术评审。软件配置审计作为正式技术评审的补充,评价在评审期间通常没有被考虑的软件配置项的特性。

状态报告

配置状态报告通过支持创建和修改纪录、管理报告软件配置项的状态或需求变化并审核变化来实现,它提供用户需要的功能,跟踪任意模式的软件项,提供完整的各种变化的历史版本和汇总信息。配置状态统计涉及记录和报告变更过程的状态,其目的是为了持续地记录配置状态以及保持基线产品和其变更建议的历史,并使相关人员了解配置和基线状态。当一个软件配置项标识更改,或变更控制审核人批准一次变更,则生成一个配置状态报告。它包含在整个软件生命周期中对基线所有变更的可跟踪性报告,主要描述配置项的状态、变更的执行者、变更时间和有何影响。配置状态报告的结果存入数据库,管理者和开发者可以查询变化信息,也可以评价变更。软件工程师可以看到都作了哪些修改或者每个文件都包含在哪个基线中,项目经理可以跟踪详细的问题报告和各种其他维护活动最简单的报告,应包括事务日志、变更日志、配置项增量报告等。配置状态报告对于大型软件开发项目的成功起着至关重要的作用,它提高了所有开发人员之间的通信能力,避免了可能出现的不一致和冲突。