

基于设计体模型的产品数据管理技术*

郭阳 杨军 陈旭灿 李思昆

(国防科技大学计算机系 长沙 410073)

摘要 针对电子系统和设备设计的需要, 本文提出一种面向产品对象的数据模型——设计体模型。阐述了产品数据的配置管理和版本管理的实施策略。

关键词 并行工程, 数据模型, 配置管理, 版本管理

分类号 TP391.73

Product Data Management Technique Based on Design Body Model

Guo Yang Yang Jun Chen Xucan Li Sikun

(Department of Computer, NUDT, Changsha, 410073)

Abstract According to the requirement of design of electronic system and device, this paper proposes an object oriented product data model— design body model. Based on the model, the paper presents the implementation strategies of configuration management and version management of product data.

Key words concurrent engineering, data model, configuration management, version management

大多数电子系统和设备是由电子、机械、甚至微波等部分组成, 这种多领域产品的设计包括了电子设计、机械设计甚至微波设计等多个不同的工程领域的高度一致和协同。支持并行工程的多设计领域产品数据管理集成框架已成为电子系统和设备设计自动化的主要需求^[1]。而建造这种多领域产品数据管理集成框架的关键是采用灵活、有效的产品数据表示模型。

对于产品对象而言, 设计信息不仅量大而且涉及多个领域, 数据管理有着特殊的要求和特点。人们探索了许多适于产品数据管理的数据模型^[2~4], 这些模型在不同程度上可满足多领域设计的需要, 但也存在一些明显不足, 主要表现在: 它们往往不够全面和灵活, 缺乏将多领域数据统一表示和管理的方法; 它们大都能较好地支持 bottom-up 设计, 而对 Top-down 设计的支持不够灵活和自然, 很难满足大型电子系统和设备设计的需要。

针对上述问题, 我们提出了一种面向产品对象的数据模型——设计体模型, 在设计体模型的基础上, 实现了产品数据的配置管理和版本管理。

1 设计体模型

定义1.1 设计项目是与特定产品相关的所有工程设计活动的总称。根据实际应用的复杂度, 可将特定产品的设计分解为多个部件(或组件)的设计, 这些具有一定功能的部件称为设计对象。

定义1.2 一个设计对象可以由其它设计对象组成, 一个设计项目中的各设计对象形成层次结构。层次中的根设计对象称为设计主体, 其它设计对象称为设计实体。

* 国家863高技术课题资助

1997年10月5日收稿

第一作者: 郭阳, 男, 1971年生, 博士生

设计主体是对工程设计中产品对象的抽象描述,它是一个相对的概念,随着具体设计任务的变化而变化,在概念上它强调的是产品数据的总体性。而设计实体是对产品对象不同设计层次的设计单元的抽象描述,概念上设计实体可以理解为有关产品对象的局部化相关数据的集合,具有局部的独立性,与特定的设计主体相联系。

定义1.3 对设计主(实)体进行实例装配以后所产生的设计对象称为设计实例。一个设计实例可由五元组(项目名,对象名,视图名,实例名,版本号)唯一确定。其中视图是对产品设计不同方面的一种表示,如电子设计一般有为行为、结构、物理几种视图。以上设计主体、设计实体及其设计实例统称为设计体。设计体模型的概念框架如图1所示,该模型能清楚地描述产品对象的层次分解关系及属性版本关系。

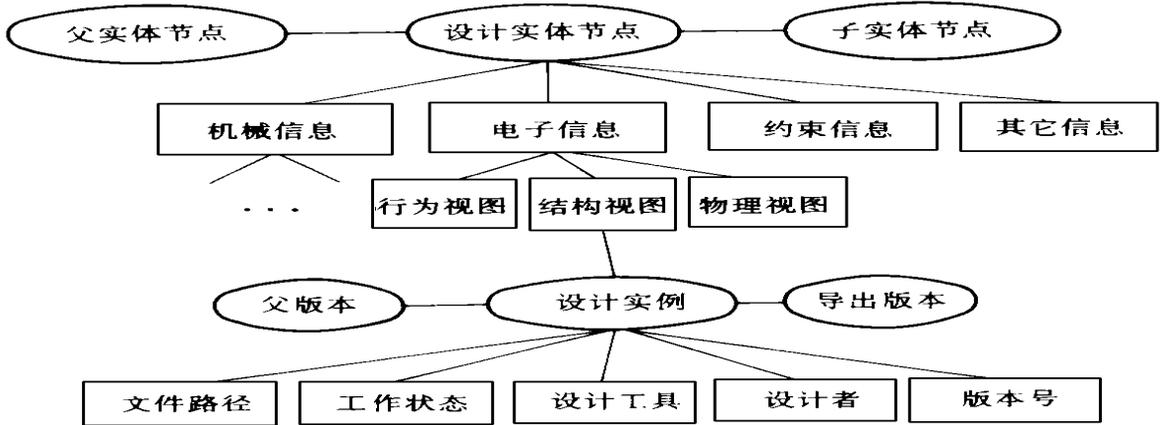


图1 设计体模型的概念框架

(1) 层次分解关系

设计实体分为元设计实体和复合设计实体两种。元设计实体代表了基本层次上的设计单元,复合设计实体则由基本单元组合而成。设计主体和复合设计实体可以引用若干个具有相关外部接口的下层设计实体,由此构成设计主体到元设计实体的多层结构。

(2) 属性版本关系

每个设计实体节点包含四类属性信息:

机械信息: 含几何图形、材料、工艺三种视图;

电子信息: 含行为、结构、物理三种视图;

约束信息: 描述设计实体的各子实体间的结构关系约束和时间关系约束,这些约束信息可用来进行自动约束检查;

其它信息: 主要是一些管理信息,如产品的描述与标识、产品的结构及声音、图像等附加说明信息。

在产品的生命周期里,项目管理者关心整个模型,而设计者关心某个实体节点。每个设计实体的信息可由一个或多个具有层次关系的设计实例表达。若某类设计实例有多个版本,则系统依据版本模型描述并维护各版本之间的依赖关系。

设计体模型的结构有以下特点: 具有直观性,反映了产品的结构本质;所有产品通过内在联系集成起来,为集成化管理提供了可能途径;可以描述完整的产品数据,覆盖了产品设计的各个阶段的全面信息,使机械与电子设计信息能集成在同一模型中。

产品数据管理采用以设计体模型为中心的设计方法。产品设计过程管理、配置管理和版本管理都围绕设计体模型进行。下面将主要介绍配置管理和版本管理的实施策略。

2 配置管理

2.1 产品结构管理

在产品的概念设计阶段,设计者首先要对产品的各主要部件按功能不同进行划分,并据此建立产品的设计体模型。产品结构管理用于建立设计体模型。设计体模型只是对产品结构的抽象描述,产品数据管理允许用户对设计体定义属性类。设计过程中产生的关于设计体的设计数据分属于设计体的某一类属性。属性类之间可以有层次关系,但只有位于叶节点上的属性类才指向设计数据。设计体的设计数据以文件的方式保存在数据库中,配置管理支持多方案的设计管理,设计体的属性类可指向设计数据的多个版本。除版本号外,设计数据的完成时间也是产品配置的一个重要规则。因此,版本管理不仅要建立设计数据的引用关系,还应将设计数据的完成时间和版本号保存为设计数据的属性。某一设计实体的属性,也可分为复合属性和元属性。元属性指基本层次上的属性,它对应于具体的设计数据;复合属性是由元属性组合而成的类树形结构。在进行产品配置时,只能按元属性进行配置。

在确定设计实体时,可以在实体(主体)树中选择适当的结点,根据使用者的权限,可以在该结点上添加新下级实体或删除已有下级设计实体,并可调出所选中的实体结点属性窗口,编辑该结点的属性树,将具体的设计数据文件与元属性建立联接关系,也可以在属性窗口中打开产品配置器,对选中的元属性结点进行产品数据配置。

产品数据管理采用 STEP 标准的 EXPRESS 描述语言和可视化建模工具(VMT)两种方式建立产品结构模型。Express 解释器将 EXPRESS 语言描述转换成 C++ 类定义和数据库操作,并将产生的模型层次信息存入数据库中。而可视化建模工具支持用户以直观的图形建立和编辑模型,它把模型的结构化信息以一种图形化、可编辑的方式提供给用户,使其对模型的改变和更新非常方便。系统支持图形和 EXPRESS 语言的相互转换。

2.2 产品配置管理

产品配置管理提供自动配置和手工配置两种方式。配置的规则可按设计数据的元属性类、设计数据产生的时间及版本号来指定。配置规则包括时间和版本两种。对于时间规则,有限定时间段、最近某时间两种;版本规则有限定某版本和不限版本两种。系统根据配置规则对以该设计主体为根的设计实体树进行自动配置。

配置管理器提供产品结构模型浏览器,支持用户自己对产品的设计数据进行配置。手工配置需要使用者提供进行配置的设计主体以及元属性,系统将列出该设计主体的直接下级实体的所选元属性不同版本的设计数据,由使用者进行选择。并可进入设计主体的某一直接下级实体的配置窗口,对下级实体进行配置。

3 版本管理

3.1 数据的组织与存储

按照配置管理定义的设计体模型和产品结构,设计实体可有多个属性类,设计过程中产生的数据保存在设计体相应的属性中。中心数据库存放系统的高层管理数据和工程设计阶段性成果数据,各子数据库管理各工具运行时的数据。按照设计过程的定义,在各设计步开始执行时,各工具从服务器上读取相关的数据信息。在各设计步执行时,数据由各工具的子数据库或工具本身进行管理,并保存在设计者所在的客户端。当该设计步执行结束时,由设计者向服务器提交数据,服务器按照设计体模型在相应的属性类中填写信息。服务器上的数据可供有授权的用户共享。

3.2 数据的安全机制

数据的状态有:设计状态、审核状态和完成状态。处于设计状态的数据是不稳定的,可以被更新和删除,表明数据仍在设计中,是不完全的;处于审核状态的设计数据是相对稳定的,表明设计实体的某设计步的初步设计已经完成,等待审核;处于审核状态的设计数据不能被更新,但可以被删除;数据处于完成状态,表明设计已通过审核,符合设计要求,该状态下的数据不允许被更新和删除,但可以被所有授权用户所共享。为了保证数据的安全,我们在两个层次上采取了保护措施。首先,根据版本的状态设立不同的处理权限,系统根据版本的状态控制对版本的操作;其次,对服务器上的数据设立显示的权限控制,通过

设置口令使项目中处于不同地位的人员对服务器上的数据具有不同的处理权限。

3.3 功能函数

为实现数据的统一管理,也便于分布式集成框架中各子系统访问数据,我们设计了一个统一的接口,提供一组基本功能函数,各子系统可按照各自的需要,设计自己的数据视图来访问服务器上的数据;另外,提供数据浏览功能,工程中处于不同地位和角色的人员可根据自己的需要配置数据。初步实现的功能函数有:提交版本,删除版本,浏览版本,修改版本参数。

3.4 版本管理

版本管理与设计过程管理密切相关。它根据设计过程任务的定义和实现,跟踪版本的历史进化过程。系统约定对应设计流的每一数据采样点,在服务器上保存各子项目的最新完成状态或审核状态的设计数据版本,而设计状态的设计数据版本原则上采用生成该数据的CAD工具的版本管理机制,但用户也可将其提交服务器管理。对应数据的状态定义,版本的状态相应也分为设计,审核和完成三个状态。在各CAD工具的各自版本管理基础上,版本管理完成以下功能:

(1)版本的生成和更新:在某一设计步中(如逻辑设计),版本按线性关系进化。在设计数据版本尚处于设计状态时,版本的历史和进化由各工具的版本控制机制来跟踪,用户也可将数据提交给服务器保存。每次工具运行结束时,将工具在该路径下生成的最新数据复制到服务器相应路径下,并记录该数据生成的时间、标记版本的状态,有前序设计步的数据则保留使用的输入数据的版本和时间。

(2)版本的删除:单一版本的删除仅与版本的状态有关,服务器根据版本的状态和删除者的权限确定版本能否被删除;复合版本的删除按照版本的结构模型,将删除其所有组成版本;父版本的删除将引起子版本的删除,而子版本的删除不影响父版本。因服务器上的数据都是设计阶段性的成果,除设计状态的版本可直接删除外,其他所有的删除都必须是授权删除。

(3)维护版本一致性,设计进入有前序设计步的工具时,若是第一次,则读取服务器上的紧前设计步的数据;否则,检查服务器上该工具的数据使用的关于紧前设计步的数据的有关信息(如版本名、版本号和时间)与服务器上保存的紧前工具的完成状态的版本名和时间是否一致,是则完成一致性检查并进入当前设计步所选工具,否则通知设计者前面数据已有修改,提请设计者注意转换。按复合版本关系图,每个子版本必须在父版本是完成状态,至少是审核状态下才能产生。

(4)基于设计流程,设计体模型绘出版本的结构关系图,静态地描述版本的结构。

(5)按照设计流程的实例运行,按设计步分阶段建立版本演化模型、传播模型,动态地描述版本的变化,绘出版本历史进化图。

4 结束语

基于设计体模型,我们开发成功了一个支持电子设备整机并行设计的产品数据管理集成框架,在实际设计中显示出良好的特性。我们将进一步把设计体模型与过程管理及项目管理相结合,使模型更为准确,更具通用性。

参考文献

- 1 全春来,李思昆等.电子系统和设备并行设计集成框架.见:全国第八届CAD与图形学学术会议论文集.青岛,1996:394~404
- 2 Wagner F R et al. Design Version Management in the GARDEN Framework. In: 28th ACM/IEEE Design Automation Conference, 1991: 704~710
- 3 Schettler O et al. BEPPO: A Data Model for Design Representation. In: Europe Design Automation Conference, 1993: 378~382
- 4 谭向东等.适用于VLSI数据管理和工具控制的面向对象方法.计算机辅助设计与图形学学报,1996,8(2):135~142