

基于 Teamcenter8 平台的元件汇总表功能开发

李 登

(西安导航技术研究所, 西安 710068)



摘要: 论述了一种基于Teamcenter8平台的元件汇总表开发方式, 此开发较为灵活且可配置, 可以帮助设计人员快速生成准确可靠的元件目录, 同时可以应对多变的规则变化。

关键词: PLM; Teamcenter8; 电气装配结构; POI; Eclipse

Function Development of Element Directory List Based on Teamcenter8 Platform

Li Deng

(Xi'an Research Institute of Navigation Technongy, Xi'an 710068)

Abstract: This article focuses on a more flexible and configurable element directory development, which is based on the Teamcenter8 platform. It can help electrical engineers to quickly generate accurate and reliable Element Directory, and can also respond to changing rules.

Key words: PLM; Teamcenter8; electrical assembly structure; POI; Eclipse

1 引言

随着人类社会的数字化、信息化和自动化, 电子元器件在国防行业中的使用越来越频繁, 担负的任务也越来越重要。电子元器件的选用及装配的正确关系到整个系统能否正常运转, 几乎与器件本身的性能同等重要, 如何保证元器件数据的准确性及可靠性保证技术已经成为当前产品全生命周期管理的一个极为重要的研究方向。

产品全生命周期管理 (Product Lifecycle Management, 简称 PLM) 系统是保证数据完整、准确的管理工具, 已经在军工行业中起到了越来越大的作用, 在电气数据管理方面形成一套完整的从元器件选择和使用、元器件评审、采购、检测与筛选、供方控制、入库、保管、出库、发放, 到元器件失效分析、处理, 到完成电装和调试的采购、调试、管理体系。

但是随着电气设计、仿真技术的不断进步, 电气设计将趋向整体化、交叉化和综合化。这些变化虽然

为电气学科的发展提供了有利的环境, 推动了电气设备向智能化方向发展, 但是越来越多的生产任务以及交叉的业务需求变更, 对元器件的数据管理以及后端进行的采购、齐套、装配造成了很大的麻烦。为了更有效地提高数据的可靠性, 完整性, 完成元器件选用及采购的控制, 需要建立一个能够自动控制的机制来完成这一功能目标。

一般在电路设计软件中, 都有元器件汇总的功能, 但是通过此功能汇总出的数据并不能直接指导生产和采购, 需要根据各企业要求再次腾写到其他软件打印出来。本文通过对 PLM 系统的二次开发, 能够快捷并准确地汇总所选用的元器件。

2 Teamcenter8 平台系统架构简介及所遇到的问题

Teamcenter8 平台是全球著名的服务和软件咨询公司西门子 (Siemens) 的产品。它是一个集成解决方案, 以产品为中心, 以项目管理为手段, 覆盖了产

作者简介: 李登 (1985-), 助理工程师, 计算机科学与技术专业; 研究方向: 企业信息化。

收稿日期: 2013-04-08

品生命周期的各种活动,以管理结果为中心,进行各类业务数据的管理并实现相应的管理思想。**Teamcenter8** 平台的二次开发包括两种方式:一种为服务器端 (ITK),以 C 或 C++开发为主,特点是部署方便,效率较高,通常用于较大数量数据的提取及流程的应用;另一种是客户端的开发 (RC),以 JAVA 为主,基于 Eclipse 平台进行插件开发,开发方式灵活。本文将重点关注客户端的开发如何满足元件目录表灵活的开发需求。

一般的企业在进行电路设计结果的管理时,将电路设计的完整设计工程包交设计师保存,使用时再进行调用。使用 PLM 系统管理后,设计流程变更为,设计时通过在软件中选取 PLM 系统中元器件信息组成产品结构,设计完成后打包上传,在系统中完成电子化审签后归档,在指导物资采购及生产加工装配时

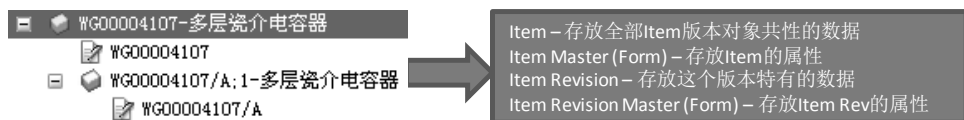


图 1 元器件 Item 基本结构

在功能开发过程中,企业会制定纸质管理时所存在的数据模板,用于统一设计规程及标准化的需要。PLM 系统可以随时随地对电子化的产品结构进行器件信息提取及汇总,高速准确反应产品结构的更改,快速指导齐套及装配工作。为了尽可能地降低业务流程变更给设计师带来的变化,将产品数据按照企业所需模板进行开发定制。

Teamcenter8 采用基于 Eclipse 平台的 RCP (Rich Client Platform) 插件进行技术开发,使用扩展点增加菜单项,通过后台调用 JAVA 程序进行处理输出,以 **Teamcenter8** 系统自带的 API 及相关 API 实现所需要的功能^[1]。

根据需要分为四个主要的功能模块进行设计:产品结构根对象及其整体子结构数据的获取,对获取到的数据处理及排序,注入 Excel 中,对于上述三种业务逻辑,可以把程序划分为“顶层产品结构获取”、“产品数据处理”、“注入 Excel”三个模块进行设计。

为了满足企业不断变化的需求及元器件标准修订带来的重复开发,系统在设计时采用了基于首选项配置的开发方式,通过调用首选项中的元器件类别条目,可根据企业需要汇总出各类元器件信息及变更元器件汇总表中的元器件类型数据,灵活地进行汇总。

根据上传的产品结构进行物资齐套及生产。在进行开发时发现很多问题,图纸信息不全导致汇总出的数据不准确,必须根据企业标准化要求进行导出,最终还是通过制度层面的约束及技术上的实现解决了这些问题。

3 实现原理

Item是**Teamcenter8**中的基本数据模型,如图1所示。一般是产品的组成部分(整机/整件/部件/零件/标准件/元器件)或者是其它的数据(技术文档,工艺规程,工程更改等)。Item下的数据一般是用于版本控制的数据,版本上的信息是可跟踪的,可恢复的,或者是用于具有产品结构(Bill Of Material,简称BOM)的产品。

4 实现方式

在 **Teamcenter8** 进行客户端的开发,包括两个步骤:程序注册以使系统识别、程序开发实现逻辑。

4.1 程序注册

程序注册的目的是就是要让 **Teamcenter8** 的平台能够识别当前开发内容。只需要在在客户化程序包 com\teamcenter\RAP\common\actions 目录下新建一个客户化的名为 actions_user.properties 注册文件,添加代码完成对程序的注册即可。

4.2 编写程序逻辑

程序注册完成后,需要在 Eclipse 开发平台下完成元件目录表程序逻辑的编制,以完成元件目录表的汇总功能。

4.2.1 顶层产品结构获取

程序的入口在客户端结构管理器菜单中,点击注册时加入的菜单“元件目录表”,自动开始工作:初始化对象 AbstRAPtBOMLineViewerApplication 与 TCSession; 首先定义对象 Tree,用于获取树状结构 BOMTreeTable Tree = app.getViewableTreeTable(), 获取 BOM 行所对应的 Item TCComponentItem relItem

=selectedLine.getItem(); 根据该 Item 的版本获取该版本的表单属性 TCComponentForm relItemVerMaster=(TCComponentForm) relItemVer.getRelatedComponent("IMAN_master_form_rev")。

4.2.2 产品数据处理

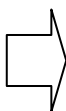
获取顶层产品结构及连接关系后,进行预数据处理。首先获取连接状态及选中的 BOM 行 YJReport yjReport=new YJReport(selectedLine,session); 接下来对数据属性进行提取,定义 Vector<vtKeys>存放数据属性名,递归获取选中行下的所有只要是元器件对象类型的对象: yjReport.getYJDatas(selectedLine.getChildren()); 然后获取器件属性: ArrayList<HashMap<String,Object>> yjDatas = yjReport.handlerYJDatas(); 最后对获取到的元器件数据进行分类,并进行处理: ArrayList<HashMap<String,Object>>yjDatas=yjReport.handlerYJDatas()。

4.2.2.1 按照系统配置的首选项对数据进行归类

首选获取元器件的分类属性,确定元器件的分类方式,获取分类属性后 itemRevision.getClassificationClass(),对前七位进行处理,然后根据“Cetc20_Class_Name_And_Id”系统首选项进行分类顺序的排序,组成 HashMap<String,Object>的成对对象,然后输出 resultDatas.put(className,arrList)。

4.2.2.2 对获取的产品结构进行排序

BOM 行	az_Boizhu
CLD102-04-000/A.1-逻辑控制板 (view)	
W600035176/A.1-尖峰抑制二极管 x 1	XT1
W600034895/A.1-连接器 x 1	Z1
W600034614/A.1-低功耗双电压比较器 x 1	V1
W600034527/A.1-集成电路 x 5	PMT2, PMT3, PMT4, PMT5, PMT6
W600033553/A.1-比较器 x 1	PMT1
W600031402/A.1-八总线收发器 x 1	D8
W600031363/A.1-四输入正或非门 x 7	D2, D3, D9, D7, D6, D5, D4
W600031204/A.1-高速双单稳态多谐振荡器 x 1	D1
W600031200/A.1-六反相器 x 1	R37



逻辑控制板						
项目代号	代号	名称、规格、型号	数量	备注	更改	
		电阻器				
C1		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C2		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C3		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C4		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C5		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C6		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C7		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		
C8		CC (U)-1-CC-50W-153T	1	5.15T 4.10T		

图 2 功能实现效果

根据获取的 arrList 属性条目,根据产品结构的 BOM 行的备注即元器件位号进行排序。首先按照位号第一位字母排序,再按照阿拉伯数字排序,如 C1~C24 等信息进行排序。最后按照 className 再次拆分为分条,并在分类中增加空行 addNullLine,组成最终所需要的对 HashMap<String,Object>。

4.2.3 注入 Excel

获取数据集对象 TCComponentDatasetType dsType=(TCComponentDatasetType)session.getTypeComponent("Dataset"); 根据数据集对象获取系统内定义的模板 TemplateDS=dsType.find(""); 获取 Excel 物理文件 template(File[] TemplateFile=TemplateDS.getFiles("")); 将属性对象依次写入 Excel: HSSFRow row= sheet.getRow((short) NUM), HSSFCell cell=row.getCell((short) 2)。

5 应用效果

本功能已经通过了一定的压力测试,在某企业曾经成功获取过 1000 个器件的元件目录表,如图 2 所示,耗时 5min,且配置灵活,只需通过修改首选项即可完成规则的修改,代替了手工编辑元件目录表时繁琐的工作量,较大地提高了采购和生产的效率。

6 结束语

在 Teamcenter8 平台的基础上,运用 Java、Eclipse、POI 等多种技术对 Teamcenter8 进行开发,实现了元件目录表的数据导入^[2]。在这个功能的辅助下,优化了原有纸质化管理时的流程,优化了产品速度,提高了采购及装配的速度,提供了统一的产品数据管理

源。

参考文献

- 1 Siemens Teamcenter. Teamcenter 2007 Getting Started with Customization. 2009
- 2 Siemens Teamcenter. Integration Toolkit Programmer Guide. 2009