

一种基于 XSLT 的系统配置设计方法

齐晓斌¹, 夏菲², 张旻¹, 林卓¹, 田丹¹

(1.西安航空计算技术研究所,西安,710119;

2.国家电网辽宁省电力有限公司辽阳供电公司,辽宁辽阳,111000)

摘要 针对嵌入式实时操作系统用户经常需要手工编辑系统配置文件的技术难点,基于配置文件 CML2 系统配置的特点,提出了一种基于 XSLT 技术的系统配置设计方法。以模型驱动架构为设计模式,把系统配置的开发核心转移到模型上,通过模型的相互转换和代码生成技术得到最终的系统配置,利用 XSD 进行模型的语义约束,再通过 XSL 语言进行目标文件的转换。改进后的系统运行效果显示:该配置方法能够针对不同平台实现软件构件的“即插即用”,做到易于维护,易于操作。

关键词 系统配置;配置菜单语言;可扩展样式表语言转换;模型;组件技术

DOI 10.3969/j.issn.1009-3516.2015.03.019

中图分类号 TP302.1 **文献标志码** A **文章编号** 1009-3516(2015)03-0092-04

A Method of System Configuration Design Method Based on XSLT

QI Xiaobin¹, XIA Fei², ZHANG Min¹, LIN Zhuo¹, TIAN Dan¹

(1.Aeronautic Computing Technique Research Institute, Xi'an 710119;2.State Grid Liaoning Electric Power Company Liaoyang Power Supply Company,Liaoyang ,111000,Liaoning,China)

Abstract: In this paper, according to the characteristics of CML2 system, a kind of system configuration design method is put forward based on XSLT technology. It is mainly based on the model driven architecture design patterns, to transfer the system configuration development core to the model, get the final system configuration through the conversion between model and code generation technology, semantic constraint model using XSD. Through the XSL language to convert the target file, It is ready to use software configuration method of software component of the "plug and play" in different platform, easy to maintain and operate.

Key words: system configuration; configuration menu language; XSLT; model; component technology

商用嵌入式操作系统除了安全可靠、性能稳定、实时性强等硬性标准外,使用门槛的高低、应用开发的难易、系统配置的难度等都会直接影响它的商业化推广以及应用。因此,嵌入式实时操作系统的配

置工具已经上升到重要地位。嵌入式实时操作系统的配置文件是非常繁琐的,如果手工设置,用户不仅要清楚项目需求,还要了解内核的工作情况。如果通过配置工具,就可以根据用户的配置,自动生成配

收稿日期:2014-10-28

基金项目:“十二五”核心电子器件高端通用芯片及基础软件产品研究计划资助项目(2012ZX01041-002-003)

作者简介:齐晓斌(1983—),男,山东潍坊人,工程师,主要从事计算机软件技术、集成开发环境技术研究.E-mail:108747116@qq.com

引用格式: 齐晓斌,夏菲,张旻,等.一种基于 XSLT 的系统配置设计方法[J].空军工程大学学报:自然科学版,2015,16(2):92-95. QI Xiaobin, XIA Fei, ZHANG Min, et al. A Method of System Configuration Design Method Based on XSLT[J]. Journal of Air Force Engineering University: Natural Science Edition, 2015, 16(3): 92-95.

置文件,无需手动修改,大大减少用户的工作量,使得用户可以有更多的时间和精力投入到业务逻辑的开发上,还可以对配置进行错误检查,从而提高系统稳定性,所以,成熟的配置工具已经成为衡量嵌入式操作系统成熟的标志之一^[1]。

针对常见嵌入式实时操作系统维护繁琐、系统配置复杂等问题,本文使用 XML 模型描述进行系统配置,利用可扩展样式表语言转换 XSLT 技术进行转换规则管理,并使用模板技术对系统配置进行了新的设计构思。

1 嵌入式操作系统的配置工具

目前,为了适应各种需求以及不同的应用场景,嵌入式操作系统都是可裁剪、可配置、可移植。操作系统的文件结构一般包括:内核文件、控制内核文件编译的配置文件,还有和微处理器相关的汇编代码,其中配置文件主要实现组件的裁剪、控制系统状态等功能。用户可以通过配置工具大大简化操作系统的裁剪、配置、移植等步骤,从而降低应用的开发难度。表 1 是几种有代表性的面向嵌入式应用的操作系统的配置介绍。

表 1 典型嵌入式操作系统系统配置支持图形的对比表
Tab.1 A typical embedded operating system configuration support graphical comparison table

操作系统名称	发布者	是否支持图形配置
OSKit	University of Utah FLUX Group	不支持,只能修改文本配置文件
eCOS	CygnusSolutions	支持,但编译后不能进行重配置
Tornado	Wind River	支持,WindConfigTM 模块提供了强大的图形化自动配置功能,能够实现 Vx-works 的图形化的配置 ^[2]
uC/OS-2	Micrium	不支持,只能修改文本配置文件
嵌入式 Linux	开源社区	支持,但是系统配置的内容非常繁琐,类似命令行式的配置

其中嵌入式 Linux 系统的配置内容非常繁琐,全部基于菜单的手动配置,类似于命令行式的配置,维护非常复杂。它的实现通过解析 config.in 文件实现,该文件基于 CML2 语言实现,并详细定义了内核中每一个功能模块和它们之间的依赖关系。

2 基于 CML2 的系统配置

CML2(Configuration Menu Language)是配置菜单语言,有一系列脚本语句组成,它由条件语句、简单语句、“source”语句和菜单块语句组成。表 2

简单说明 CML2 的一些基本语法:

表 2 CML2 的一些基本语法

Tab.2 The basic syntax of CML2

基本语法	语法含义
mainmenu_name/ prompt/	用来定义配置文件的顶层名
comment/prompt/	用来在配置的过程中显示 prompt
bool/prompt/ / symbol/	将 Prompt 显示给用户,从用户接受一个值,并将该值赋给变量 symbol/。合法的输入值是 n 或 y。这种 bool 型的语句没有默认值 ^[1] 。
hex/Prompt/ / symbol//word/	将/Prompt 的值显示给用户,并且从用户那里接受一个值,同时将这个值赋给变量/symbol/,任何二进制数都是合法的输入值。/word 是默认值 ^[1] 。
int/Prompt/ / symbol/ /word/	将/Prompt 显示给用户,从用户那里接受一个值,并且将这个值赋给变量/symbol/。/word 是默认值,任何十进制数都是合法的输入值 ^[1] 。

通过上表 CML2 的语法可以看出,基于 CML2 的系统配置需要进行大量的手动赋值、菜单翻页、合法性检查等操作,通过 CML2 的脚本配置,集成度不够高,通常还需要命令干预,其配置、编译过程还需手动完成。这种配置裁剪方法虽然简单,但容易造成目标代码中存在一定量的冗余代码,容易使得内核中存在过多的条件编译语句,这样就损害了源代码的可读性和可维护性,并导致了调试和测试的复杂性,降低了裁剪率,增加了裁剪的难度。针对以上脚本配置复杂、容易产生冗余的条件编译语句、可维护性差等缺点,基于 XSLT 技术设计并实现的系统配置工具都可以克服。

3 系统配置的新设计方法

针对第 2 节描述的基于 CML2 技术实现的系统配置不足,新设计方法利用类构件技术进行构件化改造,通过新的配置工具,使嵌入式操作系统支持不同层次的可配置性。本章讨论系统配置的新的设计方法的相关实现。

3.1 设计原理

新设计的系统配置主要利用 XSLT 转换技术和 MVC 的模式设计原理,通过 XML 对系统配置进行建模,应用 XSLT 技术将 XML 格式的模型转换为目标代码。

3.1.1 采用 XML 描述的模型

XML 是 W3C 组织制定的一种数据交换和存储的开放标准,并于 1998 年发布正式版本 1.0。它是一种元标记语言,它采用有实际意义的标签描述

数据元素。在新的设计方法中采用 XML 的元素节点进行模型描述,包含模型的各种属性以及方法。

3.1.2 采用 XSD 对模型进行约束规则管理

通过 XML Schema 定义一类 XML 文档并指导 XML 文件的生成以及 XML 文件的规范性和有效性检查。在新设计方法中,采用 Schema 技术对模型进行检查,保证其语义语法以及逻辑的正确性。

3.1.3 采用 XSLT 描述进行转换规则管理

XSLT 主要用来处理 XML 文件的转换,它具有强大的语义表达能力和易于扩展的模板功能^[3],因此,可以用于描述平台独立模型到平台相关模型的转换规则。当项目规模扩大以及涉及多种目标文件时,转换规则会相应增多,在新设计方法中内置一些常用的转换规则,并作为接口提供给用户,使得该方法具备良好的可扩展性^[4]。

3.1.4 采用 XSLT 的模板扩展

通过 XSLT 的模板技术不需要 java 代码即可以自动生成系统代码,只需要在集成开发环境的安装目录下分类放置转换脚本,不同的目标对应不同的转换脚本,在 makefile 脚本中调用它们所需要的文件即可。通过模板技术可以提高开发效率,并提高了模型驱动架构设计的复用性。

3.2 新系统配置方法的工作原理

新设计的系统配置主要利用 XSLT 转换技术和 MVC 的模式设计,通过 XML 对系统配置进行建模,后台通过独立转换库进行模型解析,只要前端模型定义清楚,作为库项目存在的控制系统基本固定,视图的呈现形式可以灵活多变,只需要进行简单的视图修改即可以实现不同的视图呈现形式,这样使得开发的重点放在在模型和后台转换程序上。模型的设计原则为精、全、活,精即粒度小,做到可配置的粒度越细越好,既而做到嵌入式操作系统的可裁剪性,这正是微内核操作系统的精髓所在;全即全面,操作系统中能够通过系统配置实现的尽量放到系统配置中,符合可裁剪性原则,也可以减小操作系统以及应用人员开发的代码负担;活即灵活多变,做到组件的配置灵活多变,可扩展,可添加,使用典型而又先进的组件技术,应用 XSLT 技术将 XML 格式的 PSM 模型转换为目标代码。基于 XSLT 技术的系统配置组成见图 1,系统配置的运行流程见图 2,各功能模块的作用如下:

3.2.1 图形模块

提供用户对操作系统的对象配置。对象配置的内容主要包括是结构配置和属性配置。图形模块有三种视图,文件视图、设计视图和 overview 视图。文件视图提供用户更改描述文件的功能;对象视图

使用 TreeView 控件显示各个组件模型的层次结构,以及对象的属性和子对象;overview 视图作为一个综括视图,提供模型配置的总体情况以及一些内存配置的基本分布情况。

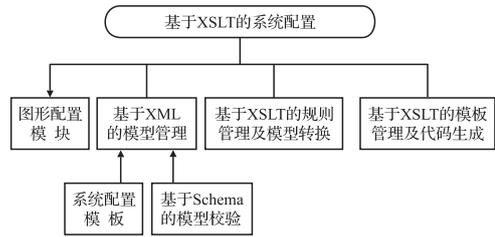


图 1 基于 XSLT 技术的系统配置的开发组成图

Fig.1 Development composition based on XSLT Technology

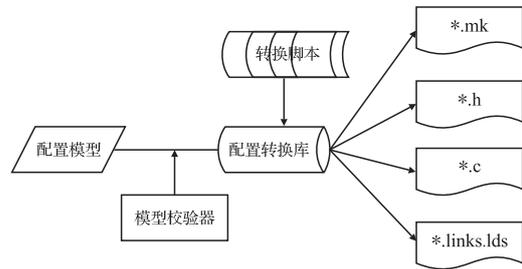


图 2 运行流程图

Fig.2 Operation flow chart

3.2.2 模型管理

模型管理模块主要包括模型的建模和解析,模型以 XML 文件存在于配置模板库中。模型包括参数、函数、事件、数据和指令五要素。参数是指模型基本配置属性,主要包括类型、名称、宏名称、对应文件类型;事件是指用于进行条件编译的执行操作序列;函数指可以被其它模型调用并完成指定任务的功能序列;指令由其他模型发出,指示接收模型完成指定任务。数据是模型自身所带有的特定数据,获取的数据可以采用缓存方式或者直接使用^[5]。模型解析把配置文件解析成 DOM 树,并向校验模块提供校验数据。

3.2.3 配置模板和模型校验

配置模板主要提供给用户一个可直接使用的模型模板库,便于二次开发,用户可以在模板之上进行再适应性修改;模型校验利用 Schema 配置文件的约束规则实现模型的有效性校验,主要包括数据类型、使用规则、内存范围三个方面的校验,以保证模型的逻辑和结构的正确性。在进行代码生成之前自动进行模型校验,并把结果实时反馈到图形界面。

3.2.4 基于 XSLT 的转换规则管理及模型转换

根据目标平台编写转换脚本规则,通过转换脚本规则生成相应的系统代码,提供最终模型。

3.2.5 基于 XSLT 的模板管理及代码生成

通过 XSLT 的模板技术只需要少量甚至不需要 java 代码即可以自动生成系统代码,只需要在集成开发环境的安装目录下分类放置转换脚本即可,不同的目标对应不同的转换脚本,在代码生成的 makefile 脚本中调用它们所需要的文件,生成的系统代码主要包括:源文件、脚本文件、链接文件三类,这样大大提高了开发效率并实现了多平台开发。

3.3 新系统配置方法的主要优势

3.3.1 业务逻辑和表现形式的分离

业务的逻辑关系和展示给用户的表现形式分离开来,易于扩展,便于维护,并且大大减少 Java 代码量,实现 MVC 架构。

3.3.2 标准技术的使用。

改进后的系统配置通过自动把资源存储到 DOM 中,然后把它们传递给视图。利用了强大的表达式语言 XPath,它可以从 XML 流中抽取数据,还使用了 W3C 标准语言,提供 CML2 配置系统所不具有的函数和递归功能。

3.3.3 清晰的界面化配置,操作简单,易于维护

改进后的系统配置界面是基于 XML 的图形化配置,通过 XSD 规则进行语义约束,用户可以进行直观的简单操作,并且模型具有模板,在用户需求进行相应变化后,可以只进行模板的修改,易于维护。

3.3.4 丰富的输出格式。

改进后的系统配置架构的一个重要特性就是能够动态的改变输出的内容类型以及用户接口的风格。通过 XSLT 技术可以获取不同的输出格式。

3.4 运行效果

通过新的系统配置设计方法实现的系统配置模块,层次化和模块化结构更加清晰,更加直观,易于操作,易于维护,大大提高了配置开发的效率。改进后系统配置界面效果对比见图 3~4。

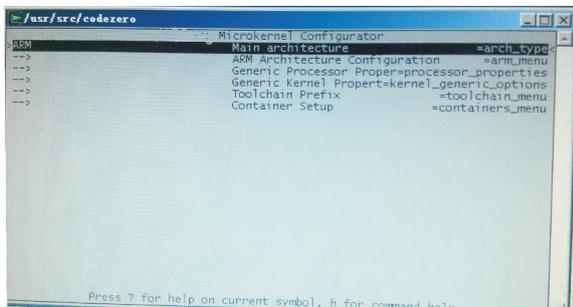


图 3 改进前界面效果图

Fig.3 Before the improvement effect drawing

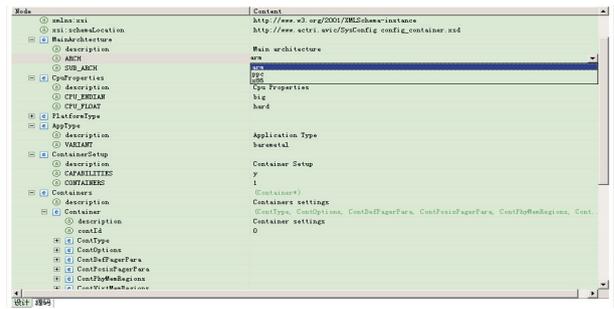


图 4 改进后的系统配置界面效果图

Fig.4 After the improvement effect drawing

4 结语

针对嵌入式实时操作系统用户经常需要手工编辑系统配置文件的技术难题,本文在简要介绍嵌入式实时操作系统系统配置的工作原理后,设计并实现了一种基于 XSLT 设计方法的系统配置工具,该工具提供了良好的软件开发环境,降低了开发难度,逐步实现了配置智能化。

参考文献 (References):

- [1] 赵幸.可定制的嵌入式实时 Linux 研究与开发[D].西安:西北工业大学,2004:12-15.
ZHAO Xing,Configurable Embedded Real-time Linux Research and Development [D].Xi'an: Northwestern Polytechnical University,2004:12-15.(in Chinese)
- [2] 冯永龙,艾明晶.Vxworks 组件技术的研究[J].计算机与数字工程,2007,35(9):62-63.
FENG Yonglong, AI Mingjing, Research on Vxworks Component Technology[J].Computer and Digital Engineering,2007,35(9):62-63.(in Chinese)
- [3] Chuank White, Mastering XSLT[J].Computer,2003,13(7):3-20.
- [4] 於良伟,袁泉,霍剑青,等.基于 XML 和 XSLT 的模型驱动构架[J].计算机工程,2010,36(9):10-13.
YU Liangwei, YUAN Quan, HUO Jianqing, et al. XML and XSLT Based on Model Driven Architecture [J].Computer Engineering,2010,36(9):10-13. (in Chinese)
- [5] 吴晓燕,吴静,陈永兴,等.基于复杂性测度的仿真模型验证[J].空军工程大学学报:自然科学版,2013,14(2):31-32.
WU Xiaoyan, WU Jing, CHEN Yongxing, et al. Validation of Simulation Models Based on Complexity Measure[J].Journal Air of Force Engineering University :Natural Science Edition,2013,14(2):31-32. (in Chinese)

(编辑:徐敏)