

# AndroMDA 模型转换组件的扩展研究

朱忠旭<sup>1,2</sup>, 袁兆山<sup>1</sup>

(1. 合肥工业大学, 安徽 合肥 230009; 2. 安徽财贸职业学院, 安徽 合肥 230601)

**摘 要:** AndroMDA 是一种支持模型驱动架构的开发工具, 模型转换组件是其实现从模型到代码的转换的核心部件。在应用 AndroMDA 进行系统开发时, 往往需要对其模型转换组件进行定制, 以生成满足特定需要的代码。文中提出了一种对 AndroMDA 的模型转换组件进行扩展的方法, 并使用该方法对模型转换组件 Bpm4Struts Cartridge 进行了扩展。使用扩展后的 Bpm4Struts Cartridge 可以从 UML 状态图生成基于 Struts 框架和 Ajax 框架 DWR 的代码, 从而实现了在 UML 模型中可视化地应用 Ajax 技术的目标。

**关键词:** AndroMDA; 模型转换组件; Struts; Ajax

**中图分类号:** TP311

**文献标识码:** A

**文章编号:** 1673-629X(2008)04-0074-03

## Research on Extension of Model Transition Component of AndroMDA

ZHU Zhong-xu<sup>1,2</sup>, YUAN Zhao-shan<sup>1</sup>

(1. Hefei University of Technology, Hefei 230009, China)

2. Anhui Finance and Trade Vocational College, Hefei 230601, China)

**Abstract:** AndroMDA is a development tool which assists MDA. Model transition component (MTC) is the core component which realizes the transition from model to code. In the development process using AndroMDA, developers often need to customize MTCs in order to generate code fulfilled demanding. Presents a method of extending the MTCs of AndroMDA, and extends the MTC named Bpm4Struts Cartridge. The extended Bpm4Struts Cartridge can generate code based on Struts framework and an Ajax framework named DWR from activity diagram, so realize the goal of applying Ajax technology visually in UML model.

**Key words:** AndroMDA; model transition component; Struts; Ajax

## 0 引言

AndroMDA<sup>[1]</sup>是遵循MDA<sup>[2,3]</sup>开发范型的一个开源的开发工具, 模型转换组件(Cartridge)是 AndroMDA 进行模型转换的核心插件, 由它来完成从模型到代码的生成任务。在使用 AndroMDA 进行系统开发时, 使用何种模型转换组件是由开发者所选择的目标平台决定的。当没有现成的模型转换组件可用时, 开发者需自己创建新的或扩展已有的模型转换组件。

## 1 AndroMDA 的实现原理

AndroMDA 导入由建模工具生成的以 XMI 文件格式表示的平台独立模型, 并在 MDR 中将其实例化为类似 DOM 树的形式; 然后对实例化后的 PIM 进行遍历, 在遍历的过程中, AndroMDA 生成相应的 Metafacade 对象, 通过该对象将 PIM 转换为相应的

PSM; 再通过模板引擎及相应的模板文件, 将 PSM 转换成代码。

AndroMDA 由模型仓库、模板引擎、转换库和模型转换组件等部分组成, 其中, 模型仓库提供了用于存储元数据的持久化机制; 模板引擎是负责代码生成的重要组件; 转换库是由多种转换方法组成的转换方法库, 它的作用是将 PIM 中的某些平台无关内容转换为平台相关的内容; 模型转换组件是 AndroMDA 用于定制模型转换的重要插件, 它的功能是对标注了范型或满足推断条件的模型元素进行处理。

## 2 AndroMDA 模型转换组件的扩展方法

在对 AndroMDA 实现原理的研究的基础上, 通过扩展 AndroMDA 模型转换组件建立了由 EModelTran、ECodeGen 和 Econfigure 构成的模型转换机制, 如图 1 所示。在这种扩展机制中, EModelTran 用来实现从 PIM 到 PSM 转换的扩展, ECodeGen 用来实现从 PSM 到代码生成的扩展, EConfigure 用来为扩展后的模型转换组件提供相应的配置信息。

收稿日期: 2007-07-09

作者简介: 朱忠旭(1974-), 男, 安徽凤台人, 硕士, 研究方向为软件工程; 袁兆山, 教授, 研究方向为软件工程、计算机网络。

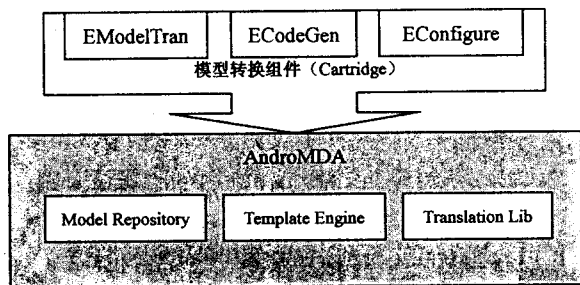


图 1 AndroMDA 模型转换组件的扩展机制

### 2.1 EModelTran

EModelTran 的建立通过以下几步来实现:

#### 1) 创建扩展部分的 PSM 元模型。

PSM 的元模型由平台中的核心概念和关键技术构成,通过对目标平台的核心概念及其表现形式的深入分析,提炼出类模型元素以及它们之间的关系,最终得出 PSM 元模型。

#### 2) 确定模型转换规则。

模型转换规则定义了 PIM 元模型元素与 PSM 元模型元素之间的映射关系,对模型转换组件进行扩展时,由于建模技术已经确定,因此 PIM 的元模型也已经确定。由于已经确定了要扩展的 PSM 的元模型,因此这时完全可以建立 PIM 元模型与 PSM 元模型之间的对应关系。

#### 3) 创建 UML 的构造型 (Stereotype) 和标记 (TaggedValue)。

在 AndroMDA 中,构造型和标记主要用于辅助 Metafacade 类的实现。为了确定要创建的构造型和标记,要对 PSM 元模型中的每个模型元素,在 PIM 元模型中寻找与之对应的模型元素信息。如果没有找到这样的信息,或者对应的模型元素信息不能够清晰完整地反映出 PSM 元模型元素的特性,此时就需要为 PSM 元模型元素定义相应的构造型和标记。

#### 4) 实现模型转换规则。

模型转换规则的实现方法在 Metafacade 对象中定义。每一个 Metafacade 对象为访问一类 PIM 元模型中的模型元素对象提供了简便易用的接口。通过这些接口,开发者可以很方便地访问到 PIM 元模型元素对象的名称、附有的构造型和标记等信息。根据这些信息,Metafacade 对象根据开发者定义的模型元素转换规则,创建与 PIM 元模型元素对象对应的 PSM 元模型元素对象。

### 2.2 ECodeGen

AndroMDA 借助模板来生成各种文件。创建 E-CodeGen 时,首先,通过分析扩展后的模型转换组件生成的各种文件与扩展前的模型转换组件生成的各种文件的差别,确定出需要修改的模板文件以及需要新建

的模板文件,然后进行模板文件的修改和创建工作。

### 2.3 EConfigure

创建 EConfigure 的工作包括修改 namespace.xml、cartridge.xml、metafacades.xml 和 profile.xml。其中:profile.xml 中定义了构造型和标记的逻辑名称与其实名称之间的对应关系;metafacades.xml 中包括了 Metafacade 类与 PIM 元模型元素之间的对应关系;cartridge.xml 文件中包括了模板文件与 Metafacade 类之间的对应关系;namespace.xml 文件中包含了与该模型转换组件相关的 cartridge.xml、metafacades.xml 和 profile.xml 三个文件的访问路径信息,以及在使用该模型转换组件进行模型转换时需要的命名空间属性信息等。

## 3 实 例

Bpm4Struts 是 AndroMDA 的一个模型转换组件,它用于将 UML 活动图转换成基于 Struts 框架的表示层代码。文中按上述方法对其进行扩展,使其可以生成基于 Struts 框架和 Ajax<sup>[4]</sup>框架 DWR<sup>[5]</sup>的代码。

### 3.1 创建 Bpm4Struts Cartridge 的 EModelTran

#### 1) 创建 DWR 的元模型。

DWR 框架的元模型如图 2 所示。DwrDeployTask 用来描述 DwrServlet 的部署描述信息;DwrConfigTask 用来描述需要调用的 Java 类的信息;Param 指定所需要的一些参数;Include 用于指定可以供客户端调用的 Java 类的方法;Exclude 用于指定不能供客户端调用的 Java 类的方法。

#### 2) 确定转换规则。

文中建立的 PIM 元模型到 PSM 元模型的映射规则如表 1 所示。

表 1 DWR 的映射规则

PIM	PSM	转换
transition	DwrDeployTask	当发生需要调用 DWR 的状态转换时,生成相应的 DWR Servlet 映射文件
	DwrConfigTask	当发生需要调用 DWR 的状态转换时,生成一个 DWR 配置文件
	DwrCreateTask	当发生需要调用 DWR 的状态转换时,为一个 Java 类生成一个 DwrCreateTask
	Param	当发生需要调用 DWR 的状态转换时,一个 Java 类对应一个 Param
	Include	当发生需要调用 DWR 的状态转换时,每一个要暴露的 Java 类方法对应一个 Include
	CallbackFunc	当发生需要调用 DWR 的状态转换时,每一个暴露的 Java 类方法有一个或多个回调函数
	Exclude	当发生需要调用 DWR 的状态转换时,每一个不需要暴露的 Java 类方法对应一个 Exclude

#### 3) 创建新的 UML 原型和标记。

定义了 <<DwrInvoting>> 原型,用于在模型转换时实现扩展部分的触发。表 2 是创建的部分标记。



各点的梯度方向,初始化参数单元集  $P = \text{NULL}$ ;

(2) 从  $D$  中随机选取 3 点;

(3) 计算这 3 点所确定的圆的参数  $p(a, b, r)$ , 然后判断法线方向与梯度之差是否在允许范围内,若是则转(4);否则转(2);

(4) 在  $P$  中寻找  $p_c$ , 满足  $\|p - p_c\| < \delta$ ,  $\delta$  是允许误差,若找到则转(6),否则转(5);

(5) 将  $p$  插入  $P$ , 令其计数值为 1;

(6) 若  $a - r < x_i < a + r$  且  $b - r < y_i < b + r$ , 则计算  $d_i$  到候选圆心的距离,若在允许的误差范围内,则将计数值加 1,遍历正方形范围内所有的点;

(7) 若该参数对应的圆上的点数  $M_p > M_{\min}$ , 转(8);否则,为虚假圆,将该参数特征从  $P$  中剔除;

(8) 检测到的为真实圆,判断已检测到的圆是否已达到规定的数目,若是,结束;否则,把圆上的点从边缘点集中去除,重置  $P = \text{NULL}$ ,转(2)。

### 3 实验结果

实验是在内存为 512MHz 的 P4 的计算机上进行的,所处理的图片是合成的圆图像(见图 2、图 4),其检测结果分别见图 3、图 5。实现语言为 VC++。



图2 原图像 图3 图2的检测结果

表1 计算速度对比

	传统 RHT 算法(ms)	改进算法(ms)
图2	362	215
图4	827	466

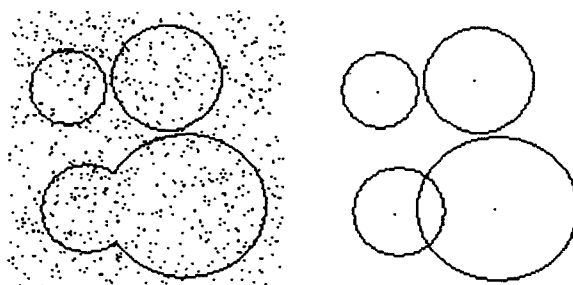


图4 带噪声图像 图5 图4的检测结果

### 4 结 语

文中的改进 RHT 算法和传统的 RHT 算法由表的速度对比中可以看出,改进的 RHT 算法在计算速度上有一定的优势。主要是由于通过对随机选取的 3 点的梯度信息判断和确定范围尽量消除无效累积使计算速度有进一步的提高。本算法对像素点多的图像处理的速度也比较快,有一定的使用价值。

#### 参考文献:

- [1] Chen Teh - Chuan, Chung Kuo - Liang. An Efficient Randomized Algorithm for Detecting Circles[J]. Computer Vision and Image Understanding, 2001, 83: 172 - 191.
- [2] Xu L, Oja E, Kultanen P. Randomized Hough Transform (RHT): Basic Mechanisms Algorithms, and Computational Complexities[J]. Computer Vision Graphics Image Process, Image Understanding, 1993, 57(2): 131 - 154.
- [3] 林金龙, 石青云. 用点 Hough 变换实现圆检测的方法[J]. 计算机工程, 2003(11): 17 - 18.
- [4] 束志林, 戚飞虎. 一种新的随机 Hough 快速圆检测算法[J]. 计算机工程, 2003(6): 87 - 88.
- [5] 张兵权, 王建峰, 卢 军. 一种新的对随机 Hough 变换改进的检测圆的方法[J]. 计算机工程与应用, 2006(18): 53 - 54.

(上接第 76 页)

项目开发时,模型转换组件的开发有着十分重要的意义。

因为模型转换组件质量的高低决定着所生成的代码的质量,因此开发模型转换组件的工作应该由项目组中水平较高的成员来完成。好的模型转换组件所生成的代码,会使用恰当的设计模式,符合最佳的设计实践,因此具有很高的质量<sup>[6]</sup>。由于项目的大部分代码都是由模型转换组件自动生成的,因此,各项目成员所交付的最终代码质量相近,风格一致。这对提高后期的测试效率,缩短项目的开发周期都有较好的影响。

#### 参考文献:

- [1] 郑 涛. AndroMDA: 免费的 MDA 午餐[J]. 程序员: CSDN 开发高手, 2004(4): 94 - 97.
- [2] 徐晓钟, 谢康林. 基于 MDA 方法学软件开发方式的原理与实现[J]. 微机发展, 2004, 14(4): 74 - 77.
- [3] OMG. MDA Guide Version 1. 0. 1 [EB/OL]. 2003 - 06. <http://www.omg.org/cgi-bin/doc?omg/03-06-01>.
- [4] Crane D, Pascarello E. Ajax in Action [M]. [s. l]: Manning Publications Co., 2006.
- [5] 柯自聪. Ajax 开发精要——案例与框架 [M]. 北京: 电子工业出版社, 2006.
- [6] 刘发贵, 胡耀民. 基于 MDA 的模式化软件设计方法与应用[J]. 计算机应用, 2005, 25(4): 790 - 791.