

实验 8 Java 代码与 UML 模型的双向转换

1. 实验目的

- UML 模型到 java 代码的转换
- 从 java 代码到 UML 模型

2. 知识要点

UML（Unified Modeling Language）统一建模语言，是一种面向对象的建模语言，它的主要作用是帮助我们对软件系统进行面向对象的描述和建模，它可以描述软件开发从需求分析到实现和测试的全过程。UML 通过建立各种类、类之间的关联、类/对象怎样相互配合实现系统的动态行为等成分（这些都称为模型元素）来组建整个模型，刻画客观世界。UML 提供了各种图形，比如用例图、类图、顺序图、协作图、状态图等，来把这些模型元素及其关系可视化，让人们可以清楚容易地理解模型。我们可以从多个视角来考察模型，从而更加全面的了解模型，这样同一个模型元素可能会出现在多个图中，对应多个图形元素。

我们可以利用面向对象建模工具，建立用 UML 描述的软件系统的模型，而且可以自动生成和维护 C++、Java、VB 等语言和系统的代码。同时，也可以利用工具，从已有的程序代码生成 UML 描述的软件系统的模型，一旦程序代码发生更改，所建立的模型就自动地同步修正。

2.1 工具的选择

1. starUML（<http://staruml.io/download>）

StarUML 是一款开放源码的 UML 开发工具，是由韩国公司主导开发出来的产品，可以直接到 StarUML 网站下载。这款用 Delphi 开发的 StarUML 曾经是最流行的开源 UML 工具，用户覆盖上百个国家，超过 350 万的下载量，但该版本从 2008 年以后没有再更新。2014 年 5 月，原开发者宣布用 HTML5, CSS, JavaScript

重新开发 StarUML 以适应更多平台，而且将走商业路线。

2. 第三方插件 Together

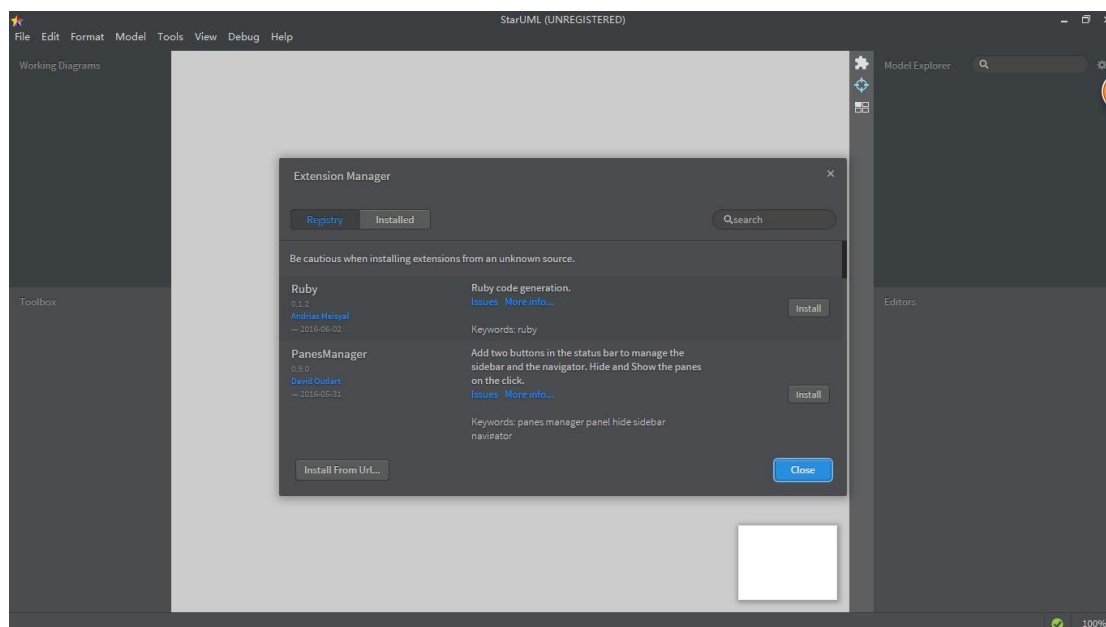
(<http://www.borland.com/Products/Requirements-Management/Together>)

Together 是 Borland 公司推出的一款将 UML 功能和 IDE 开发工具集成在一起的一套 UML 建模软件。在使用过程中可以同时看到 UML 图和 Java 源代码，修改 UML 图会使得源代码得到即时的同步修改，反过来，修改 Java 源代码也会使该代码同步地在 UML 图中反映出来。

此外，还有 Eclipse 的第三方插件 AmaterasUML（具体安装方法可百度）、MyEclipse 自带的 UML 插件、专用工具 Rational Rose 等多种工具可以供选择使用。

2.2 操作方法（以 starUML2.7.0 版本为例）

新版本的 starUML 和原来操作有些不同，需要先使用 Tools->Extension Manager 获得和目标编程语言相符合的插件。



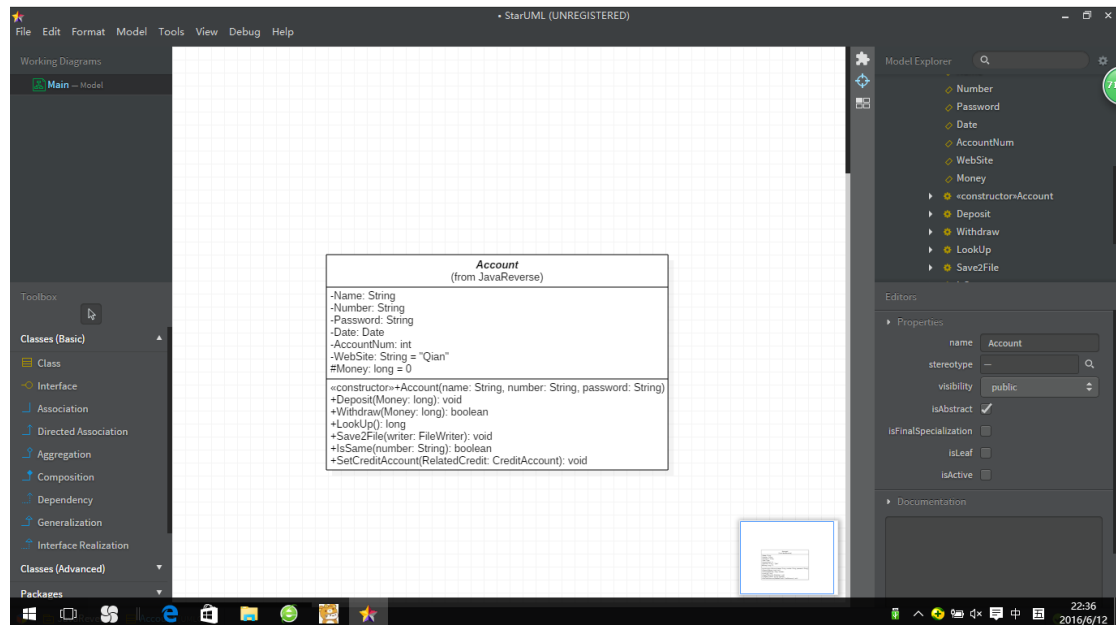
Java 插件安装成功后，再次点开 Tools 菜单会出现 java 的选项，Generate Code 表示从当前绘制的 UML 图生成 java 代码，Reverse Code 表示从已有的 java 代码生成 UML 图。

1、正向工程

在主界面下进行 UML 图的绘制,左侧导航条给出了可绘制的具体组件,单击之后鼠标拖拽到绘图区即可。具体操作方法可按照菜单提示进行。

2、逆向工程

将源代码所在的文件夹完整地导入 starUML, 根据类的层次结构可以方便地构建 UML 图。



3. 实验任务

任选一种工具, 绘制 UML 类图, 练习类图与 java 代码的双向转化。

(1) 基本功能 (用 UML 类图表示)

1) 新开账户

由 Account 类提供新开账户功能。要求储户提供姓名和身份证号, 设置密码, 添加一个新的账户, 并自动生成一个新的账户 ID, 自动记录开户日期、开户网点代号。

2) 查询账户余额

由 Account 类提供查询账户余额功能。

3) 存款, 向指定账户存入金额

由 Account 类提供向账户存入指定金额的功能, 返回操作是否成功的提示;

4) 取款，从指定账户提取金额

由 `Account` 类提供取款功能，取款操作参数为取款金额，返回实际取到金额。

(2) 向 `AccountSet` 类添加方法，实现账户数据的管理功能(用 UML 类图表示):

- 1) 保存，即当你注册了一个账户，可以将账户的全部信息进行保存。
- 2) 查询，由 `AccountSet` 类提供账户查询的功能，可以按用户的身份证号和账户的 ID 查询账户。
- 3) 删除，可以按用户的身份证号和账户的 ID 删除账户。

(3) 现需要将账户类型进一步细分，分为普通储蓄类账户和信用卡类账户。这两类账户具备相同的一些基本特征，如账号、用户名、密码、金额等。但是在操作上有所不同，储蓄类账户的金额是指账户现有余额，在取款操作中可供提取，信用卡账户的金额指账户的信用额度（能够透支的金额）。信用卡账户能够透支的金额由信用等级决定(信用等级可分为一般用户和 **VIP** 用户)。根据以上需求，请写出相应的代码框架，继承 (1) 中的 `Account` 类。观察在 UML 图中发生的变化。

(4) 现需要给系统添加一个新的功能，如果同一身份证同时注册有储蓄类账户和信用卡账号，可以实现信用卡的自动还款，即关联两类账户，并从储蓄账户中扣除相应的金额，冲抵信用卡的欠款。在 UML 图中把以上新增功能表示出来，并观察需求变更后的 java 代码。

4. 实验提交

注意：本次实验重点训练观察 UML 图和 java 代码的双向转化，把以上功能反映在 UML 图中即可，自动生成代码框架，并不要求具体实现全部代码。源程序和 UML 图打包提交。