

程序员项目开发实践系列



Delphi 7.0 + Rose 项目开发实践

吕伟臣 周 涛 主编



科学出版社
www.sciencep.com

程序员项目开发实践系列

Delphi 7.0 + Rose 项目开发实践

吕伟臣 周 涛 主 编

科学出版社

北 京

内 容 简 介

本书以 Delphi 和 Rose 软件结合的方式来阐述开发数据库项目的整个过程。书中介绍了开发数据库项目的多种方法,根据不同的项目采用 Rose 建模,书中应用的数据库是 SQL Server,也可以将其改造成其他数据库系统。本书实例取自生活中常用的几种类型:档案、教务、学生、酒店、图书、考勤、工资、库房管理等。

目前,市场上用大篇幅对 Delphi 7.0 + Rose 建模创建数据库系统的编程技术进行全面阐述的书不多,本书可以弥补上述不足,是一本主要面向中、高级编程用户的非常难得的实用参考书。

图书在版编目(CIP)数据

Delphi 7.0 + Rose 项目开发实践/吕伟臣,周涛主编. —北京:科学出版社, 2005

(程序员项目开发实践系列)

ISBN 7-03-015573-4

I.D… II.①吕… ②周… III.软件工具-程序设计 IV.TP311.56

中国版本图书馆 CIP 数据核字(2005)第 050829 号

责任编辑:吕建忠 陈砾川 / 责任校对:都 岚

责任印制:吕春珉 / 封面设计:飞天创意

科学出版社出版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

新蕾印刷厂印刷

科学出版社发行 各地新华书店经销

*

2005 年 6 月第 一 版 开本:787×1092 1/16

2005 年 6 月第一次印刷 印张:25

印数:1—4 000 字数:581 000

定价:40.00 元(含光盘)

(如有印装质量问题,我社负责调换(环伟))

销售部电话 010-62136131 编辑部电话 010-62138978-8001 (BI01)

前 言

本书作者常年从事 Borland 系列工具的程序开发工作，在长期的开发实践中，积累了丰富的经验，并收集、整理了许多宝贵的资料，现从中精选出最为基础和经典，且当前最为流行和实用的数据库部分与读者分享。

本书以 Delphi 7.0 和 Rose 项目策划软件相结合的方式详细阐述开发数据库完整项目的整个过程。书中详细介绍了 Delphi 7.0 开发数据库项目的多种方法，根据不同的项目需求，采用 Rose 建模，选定不同种类的数据库，书中应用的数据库是最为流行的 SQL Server，用户可以很容易地将其改造成其他数据库系统。在翔实的项目实例中还由浅入深地介绍了有关环境设置、PAS 语言基础知识、控件应用、技巧编程等内容，书中还给出了大量控件、API 应用技巧和许多 Delphi 的编程精华，是一本非常难得、实用的，主要面向中、高级编程用户的学习参考书。本书内容翔实，技术全面，主要以 Rose 建模与 Delphi 数据库编程应用相结合的方式引导读者学习 Delphi 7.0 数据库编程技术。目前，市场上用大篇幅对 Delphi 7.0+Rose 建模创建数据库系统的编程技术进行全面阐述的书不多，本书可以弥补上述不足。

本书的案例均经过精心挑选，力图体现 Delphi 数据库编程技术的精髓，例子之间既独立又相互联系。

本书案例包括：人事档案管理系统、学校教务管理系统、学生信息管理查询系统、酒店前后台管理系统、图书管理系统、考勤管理系统、工资管理系统、库房管理系统。

在本书的编写过程中，霍言、廉颖等同志也做了大量的工作，在此深表感谢。

由于作者水平有限，难免有疏漏之处，欢迎广大读者批评指正。

目 录

第 1 章 人事档案管理系统建模	1
1.1 系统设计	1
1.1.1 Rose 建模	1
1.1.2 创建工程的 Use Case 视图	2
1.1.3 创建工程的逻辑视图	7
1.1.4 创建数据库方案	12
1.1.5 创建组件视图	13
1.2 数据库的物理实现	19
第 2 章 人事档案管理系统的实现	23
2.1 编制数据库应用	23
2.1.1 主窗体设计过程	23
2.1.2 完善 Rose 设计出的框架结构	29
2.1.3 用户登录窗体设计	30
2.1.4 配置数据库 SetDBForm 窗体设计	33
2.1.5 备份和恢复数据 BRMainForm 窗体设计	35
2.1.6 部门管理 DepMainForm 窗体设计	38
2.1.7 系统账户管理窗体 AccountMainForm 窗体设计	43
2.1.8 人事档案管理 StuffMain 窗体设计	47
2.1.9 打印设置 PrintSetupForm 窗体设计	56
2.1.10 打印预览 PrintMainForm 窗体设计	58
2.2 运行结果	60
第 3 章 学校教务管理系统设计	64
3.1 建立教务管理系统模型	64
3.1.1 设计 Use Case View	64
3.1.2 设计工程的逻辑视图	66
3.1.3 设计 Component View	68
3.1.4 转换 Rose 模型	69
3.2 物理数据库的实现	70
3.2.1 在 SQL Server 2000 中创建数据表	70
3.2.2 配置 ODBC 数据源	71
3.2.3 采用 SQL Server DMO 创建数据库和各个数据表	72
3.3 学校教务管理系统数据库的实现	73
3.3.1 主窗体设计过程	74

3.3.2	完善 Rose 设计出的框架结构	77
3.3.3	用户登录窗体设计	77
3.3.4	配置数据库 SetDBMainForm 窗体设计	82
3.3.5	备份和恢复数据 BackupAndRestoreForm 窗体设计	85
3.3.6	班级设置 tbsClass 页面设计	88
3.3.7	课程设置 tbsCourse 页面设计	91
3.3.8	账户设置 tbsAccount 页面设计	95
3.3.9	课程查询 tbsSCourse 页面设计	98
3.3.10	成绩查询 tbsSScore 页面设计	100
3.3.11	打印设置 PrintSetupForm 窗体设计	102
3.3.12	打印预览 PrintMainForm 窗体设计	104
3.4	运行结果	105
第 4 章	学生信息管理查询系统	110
4.1	建立学生信息管理查询系统模型	110
4.1.1	设计 Use Case View	110
4.1.2	设计工程的逻辑视图	113
4.1.3	设计 Component View	115
4.1.4	转换 Rose 模型	116
4.2	物理数据库的实现	116
4.2.1	在 SQL Server 2000 中创建数据表	116
4.2.2	配置 ODBC 数据源	117
4.3	学生信息管理查询系统的实现	118
4.3.1	学生管理部分	118
4.3.2	学生选课查询部分	140
4.4	运行结果	149
第 5 章	酒店前后台管理系统	152
5.1	建立酒店前后台管理系统模型	152
5.1.1	设计 Use Case View	152
5.1.2	设计工程的逻辑视图	154
5.1.3	设计 Component View	157
5.1.4	转换 Rose 模型	159
5.2	物理数据库的实现	159
5.3	酒店前后台管理系统数据库应用实现	162
5.3.1	酒店后台管理系统	162
5.3.2	酒店前台管理系统	186
5.4	运行结果	214
第 6 章	图书管理系统	220
6.1	建立图书管理系统模型	220
6.1.1	设计 Use Case View	220

6.1.2 设计工程的逻辑视图	222
6.1.3 设计 Component View	226
6.1.4 转换 Rose 模型	227
6.2 物理数据库的实现	228
6.3 图书管理系统应用实现	231
6.3.1 主窗体设计过程	231
6.3.2 完善 Rose 设计出的框架结构	237
6.3.3 用户登录页面设计	237
6.3.4 注册用户页面设计	242
6.3.5 借还图书页面设计	245
6.3.6 图书管理页面设计	248
6.3.7 图书查询页面设计	252
6.3.8 借书类型页面设计	254
6.4 运行结果	257
第 7 章 考勤管理系统	262
7.1 建立考勤管理系统模型	262
7.1.1 设计 Use Case View	262
7.1.2 设计工程的逻辑视图	265
7.1.3 设计 Component View	268
7.1.4 转换 Rose 模型	269
7.2 物理数据库的实现	271
7.2.1 创建各数据表	271
7.2.2 配置 ODBC 数据源	273
7.3 考勤管理系统数据库应用实现	274
7.3.1 主窗体设计过程	274
7.3.2 完善 Rose 设计出的框架结构	278
7.3.3 数据库连接设置窗体设计	279
7.3.4 考勤系统登录 loginForm 窗体设计	282
7.3.5 账户管理窗体 AccountForm 设计	285
7.3.6 考勤记录 KaoQingMainForm 窗体设计	288
7.4 运行结果	299
第 8 章 工资管理系统	302
8.1 建立工资管理系统模型	302
8.1.1 设计 Use Case View	302
8.1.2 设计工程的逻辑视图	304
8.1.3 设计 Component View	307
8.1.4 转换 Rose 模型	309
8.2 物理数据库的实现	310
8.2.1 创建各数据表	310

8.2.2	配置 ODBC 数据源	314
8.3	工资管理系统数据库应用实现	314
8.3.1	主窗体设计过程	314
8.3.2	完善 Rose 设计出的框架结构	319
8.3.3	数据库连接设置窗体设计	320
8.3.4	工资管理系统登录 loginForm 窗体设计	323
8.3.5	工资账户管理窗体 AccountForm 设计	327
8.3.6	工资标准设置窗体 baseSetup 设计	329
8.3.7	工资统计 gzglForm 窗体设计	333
8.3.8	工资查询 SearchForm 窗体设计	335
8.3.9	奖励记录窗体设计	337
8.4	运行结果	341
第 9 章	库房管理系统	344
9.1	建立库房管理系统模型	344
9.1.1	设计 Use Case View	344
9.1.2	设计工程的逻辑视图	348
9.1.3	设计 Component View	352
9.1.4	转换 Rose 模型	354
9.2	物理数据库的实现	356
9.2.1	创建各数据表	356
9.2.2	配置 ODBC 数据源	357
9.3	库房管理系统数据库应用实现	358
9.3.1	主窗体设计过程	359
9.3.2	完善 Rose 设计出的框架结构	362
9.3.3	数据库连接设置窗体设计	363
9.3.4	库房管理系统登录 loginForm 窗体设计	366
9.3.5	库房系统账户管理窗体 AccountForm 设计	370
9.3.6	物品编号 ItemForm 窗体设计	373
9.3.7	货物查询 SearchForm 窗体设计	376
9.3.8	出库/入库操作 kfglForm 窗体设计	378
9.4	运行结果	390

第 1 章 人事档案管理系统建模

人事档案管理主要用于员工个人资料的录入、变动，以及企业人事结构的改动等管理工作，与传统利用人力的档案管理方法相比，可以更好地解放人力资源，同时便于领导层根据需要快速地做出相应的人事决策。

1.1 系统设计

任何数据库系统在设计前都要进行系统功能分析。本系统的设计目的主要是实现单位的人事档案管理工作。根据人事档案管理工作本身的特点，基本上可以将该系统分为两个功能，即人事档案管理功能和系统数据库维护功能。这两个功能所涉及的数据库包括人事档案数据库、与之关联的部门数据库以及保存登录账户信息的账户数据库。下面根据此特点对系统进行设计和开发。

1.1.1 Rose 建模

启动 Rose，打开 Create New Model 对话框（如图 1.1 所示），用户可以根据将要设计的项目的具体情况来选择相应的模型。例如，如果决定采用 Visual C++ 开发环境，则可以选择 VC6 MFC 6.0 模型。本例不选择任何模型，单击 Cancel 按钮。

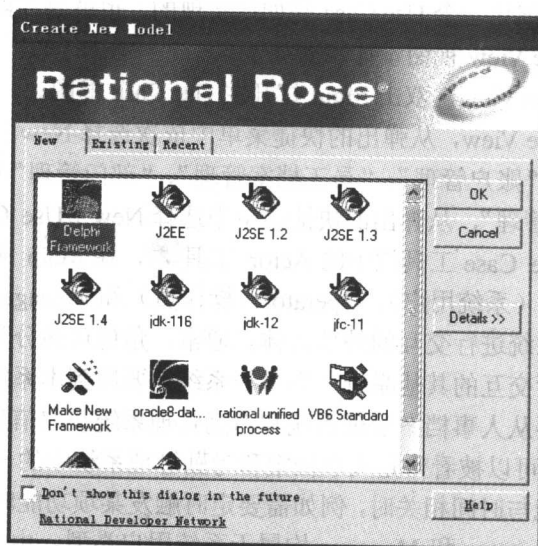


图 1.1 Create New Model 对话框

Rose 的设计窗口由 Use Case（使用案例）视图、Logic Case（逻辑案例）视图、

Component（组件）视图和 Deployment（部署）视图（如图 1.2 所示）组成。

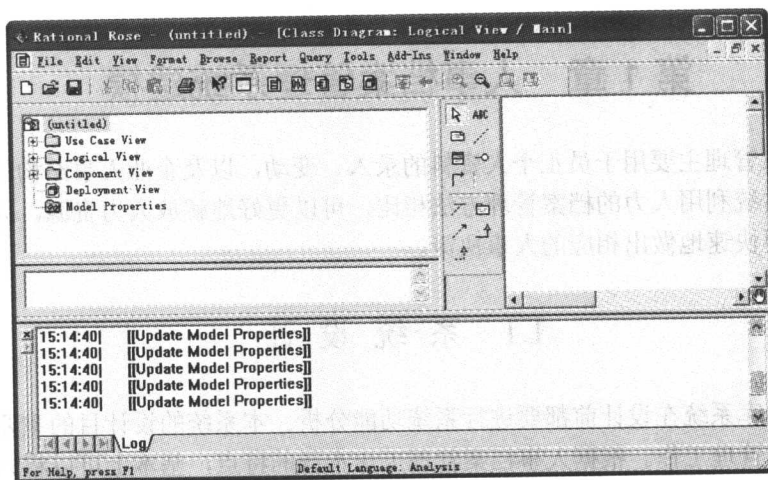


图 1.2 Rational Rose 的主窗体

1.1.2 创建工程的 Use Case 视图

Use Case 视图使用案例视图，主要被用来为分析人员、项目管理人员以及客户提供要开发系统的框架结构视图，即 Use Case 视图为设计人员提供了要开发软件的系统功能，而并不涉及系统的具体实现。实际上，一个 Use Case 就是系统要完成的一项功能，即用户如何使用该系统。所以系统要实现的那些功能，就是在该系统的设计中应该具有的那些 Use Case。相应地，细化一个 Use Case 是如何实现的，也就是在细化一个功能是如何实现的。本项目的 Case View 视图设计步骤如下：

（1）展开 Use Case View，双击 Main 打开主 Use Case 框图。

（2）右击 Use Case View，从弹出的快捷菜单中依次选择 New | Package，创建四个包，分别将它们命名为“账户管理”、“员工档案管理”、“部门管理”和“系统数据管理”。

（3）右击“账户管理”，从弹出的快捷菜单中选择 New | Use Case Diagram，将其命名为 Main。选择 Use Case 工具栏中的 Actor 工具 ，在 Main 使用案例框图中添加三个角色：SystemUser（系统用户）、Operator（操作员）和 Manager（管理者）。

角色就是与所建系统进行交互的外部实体。通常，角色可被分为三类：使用系统的系统用户、与系统进行交互的其他系统（当另一系统需要用到本系统时，例如随后介绍的工资管理系统就需要从人事档案管理系统和考勤管理系统获取信息，所以从某种意义上讲，工资管理系统也可以被看作是人事档案和考勤管理系统中的一个角色），以及时间（当系统中的某些功能与时间相关时，例如需要定时触发某项功能时）。这里添加的三个角色：SystemUser、Operator 和 Manager 均属于系统用户类型。由于操作员、管理员具有不同的系统使用权，所以在此分别进行创建。而这两种角色又具有共性，即都是所建系统的使用者，所以可将它们归于 SysterUser 类型。换句话说，这里的 SysUser 角色是一个包含所有使用该系统角色的共性的抽象角色。

(4) 单击 Use Case 工具栏中的 Use Case 工具 ，在主框图中添加一个新的使用案例，并将其命名为 Login。

(5) 为角色和使用案例添加关系。选择 Use Case 工具栏中的关联工具  (Associations, Rose 中用箭头来表示角色与使用案例之间的通信关系)，在角色 SystemUser 和使用案例 Login 之间添加关联，如图 1.3 所示。

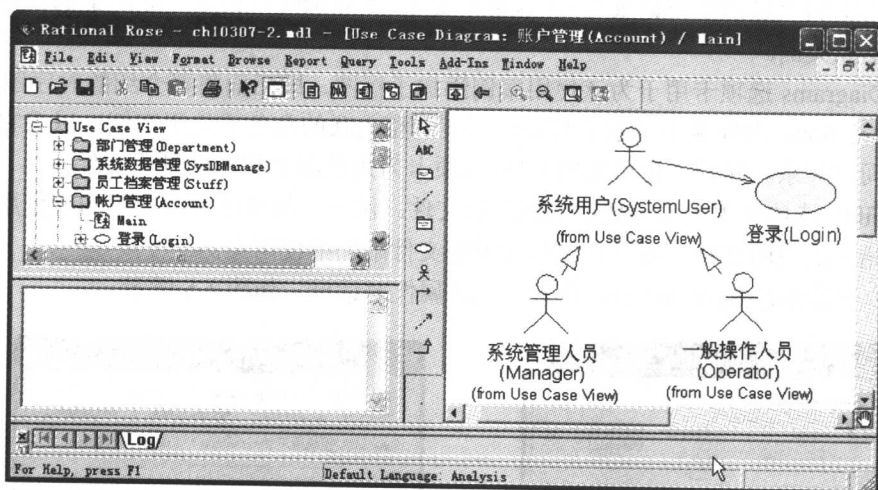


图 1.3 在角色和使用案例之间添加关联

说明：在 UML 中对使用案例和角色间定义了四种关系：通信关系、使用关系、扩展关系和角色一般化关系。其中通信关系描述了角色与使用案例之间的关系；使用关系和扩展关系则用于描述使用案例和使用案例之间的关系；而角色一般化关系则用于描述角色之间的关系。

(6) 在文档窗口中为使用案例 Login 添加文档说明，“操作人员是否能够登录系统，以及登录后所拥有的权限”。文档说明不会影响最终生成的具体编码。

选择工具栏中的 Generalization（用于表示使用案例与使用案例之间的关系时，代表使用或扩展关系；用于表示角色与角色之间的关系时，则代表角色的一般化关系）工具，分别从 Manager 和 Operator 向 SystemUser 连线。角色一般化关系，便于理解各角色之间的关系。本例很清楚地表示了 SystemUser、Manager 和 Operator 之间的关系。

建立好基本角色、使用案例以及它们之间的关系后，分别查看它们的属性。

(7) 双击使用案例 Login，打开其 Specification（规范）对话框，如图 1.4 所示。从中可以了解到与使用案例 Login 相关的规范和属性。

- 在 General 选项卡下的 Name 输入框中可以定义使用案例的名称，对于本使用案例，将其命名为登录(Login)。如果需要可在 Stereotype 中选择使用案例的版型。版型就是类型，如果觉得有必要将使用案例分为多种不同的类型，就可以使用该选项来划分使用案例的版型。Rose 为用户提供了一些自带的版型，用户也可以直接在输入框中输入自己命名的版型名称。通常不对使用案例划分版型，而是在 Logical View（逻辑视图）中对相应的类或操作划分版型。这里没有对 Login

使用案例划分版型。Rank 用于指定该使用案例的优先级，当此项目比较大而复杂时，随着项目的进展可能需要了解处理使用案例的顺序，此时可以为使用案例设置优先级别。本使用案例没有设置优先级别。Abstract 用于指明该使用案例是否为抽象案例，抽象案例是指那些不是由角色直接使用的案例，例如在随后介绍的“修改账户记录”和“删除账户记录”两个使用案例，需要使用另一个使用案例“显示账户记录”，在这种情况下，“显示账户记录”使用案例就是一个抽象案例。

- Diagrams 选项卡用于为该使用案例添加子框图，即后面将要介绍的时序图。
- Relations 选项卡中显示了与该使用案例相关联的角色或使用案例，以及它们之间的关系。此外，也可使用 Files 选项卡为角色添加相应的说明文件。
- Files 选项卡用于为该使用案例添加文件。例如，如果已经编写了与该使用案例相关的说明文档，可以在此将该说明文件附加给使用案例。

(8) 双击角色 SystemUser，打开其规范属性对话框，如图 1.5 所示。

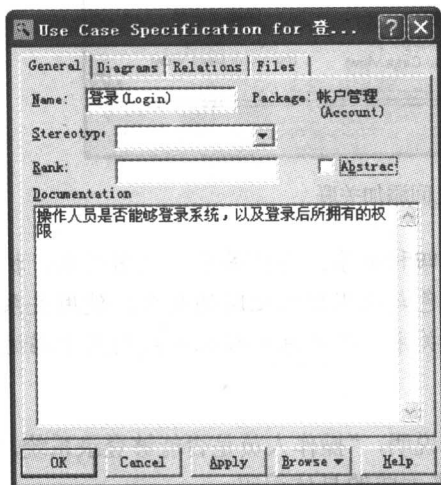


图 1.4 使用案例 Login 的 Specification 对话框

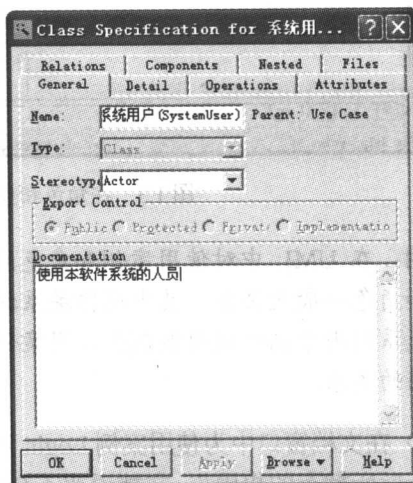


图 1.5 角色 SystemUser 的规范属性对话框

- General 选项卡：用户可以设置该角色的名称、版型以及添加说明信息等。注意，如果更改了角色的版型，Rose 将不再使用标准的 UML 符号来表示该角色。
- Detail 选项卡，可以根据实际需要设置 Multiplic (数量) 参数，对于某些角色来说可以是多人，例如 Operator，而对于某些角色来说则可能只允许 1 人。在此将 Operator 的 Multiplic 参数设置为 n(如果确定只允许一位人员来管理系统数据库，那么也可在此将 Manager 的 Multiplic 参数设置为 1)。

说明：如果想删除 Use Case View 中的角色，使用案例或关系，可选中它，然后按下 Ctrl+D 键，就可彻底地将相应的对象从模型中删除。

(9) 在“账户管理”包下的 Main 使用框图内继续添加使用案例：AddNewAccount

(添加新账户)、ModifyAccountRecord (修改账户信息)、DeleteAccountRecord (删除账户记录)、ShowAccountRecord (显示账户记录), 并在角色 Manager 和这些使用案例之间添加通信关系。此外, 如果想要修改或删除一个账户, 首先需要显示账户记录集, 然后选择需要修改或删除的账户记录, 最后才能执行修改或删除操作。所以, 通过分析可以知道, ModifyAccountRecord 和 DeleteAccountRecord 这 2 个使用案例与使用案例 ShowAccountRecord 之间存在着使用关系。选择工具栏中的 Generalization 工具从 ModifyAccountRecord 使用案例到 ShowAccountRecord 使用案例间, 以及 DeleteAccountRecord 到 ShowAccountRecord 之间连线。“账户管理”包的 Main 使用框图如图 1.6 所示。

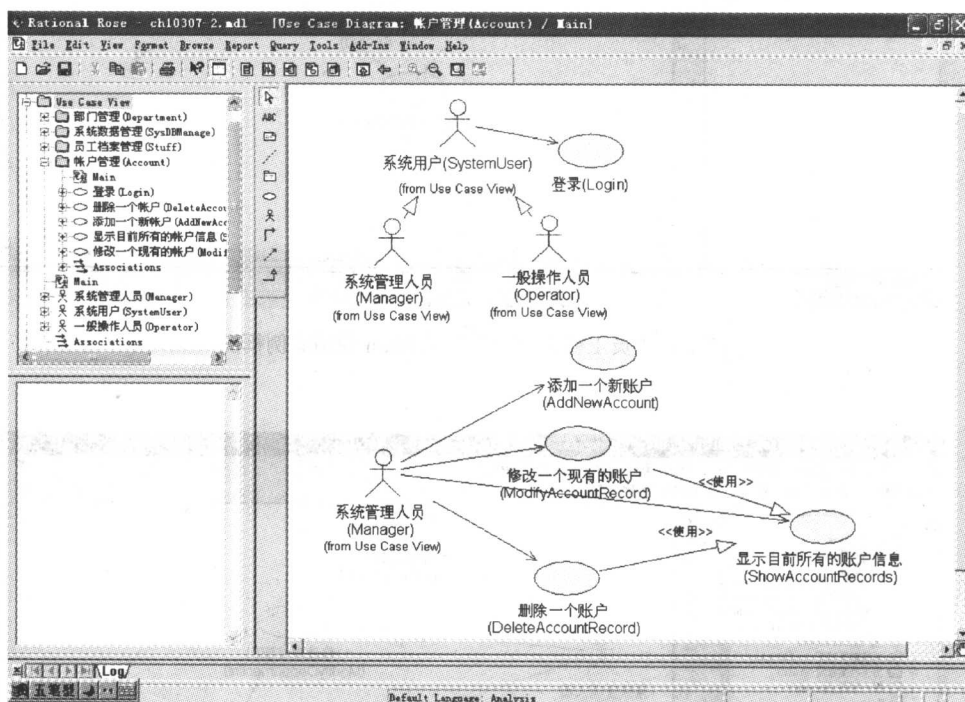


图 1.6 “账户管理”包的 Main 使用案例框图

(10) 继续为“员工档案管理”包添加 Main 使用案例框图, 然后添加相应的角色和使用案例, 如图 1.7 所示。

(11) 为“部门管理”包添加角色、使用案例及它们之间的关系, 如图 1.8 所示。

(12) “系统数据管理”包的 Main 使用案例框图如图 1.9 所示。

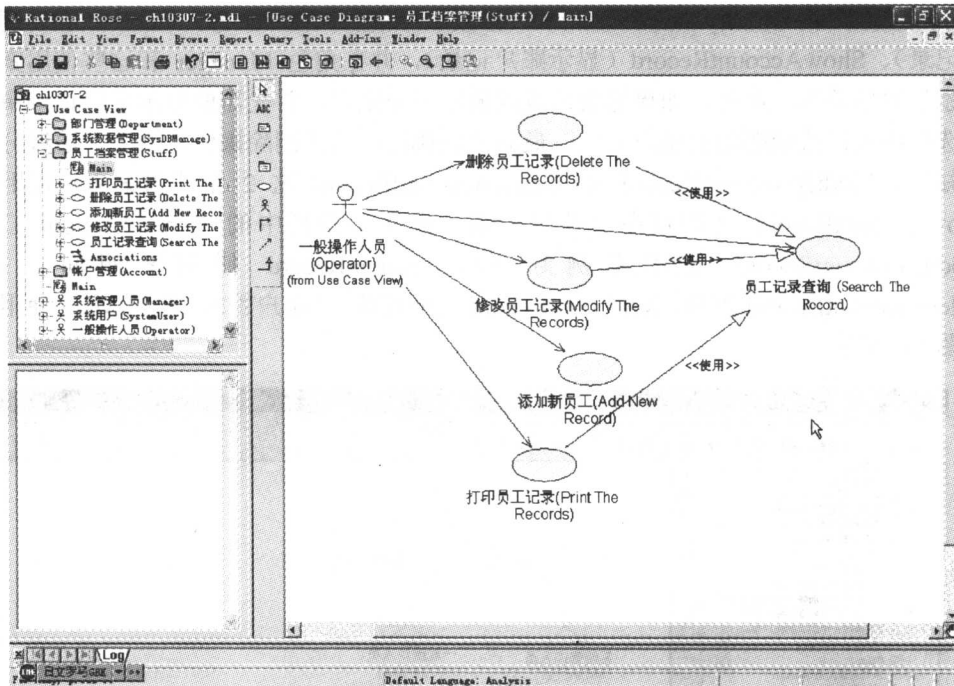


图 1.7 “员工档案管理”包的 Main 使用案例框图

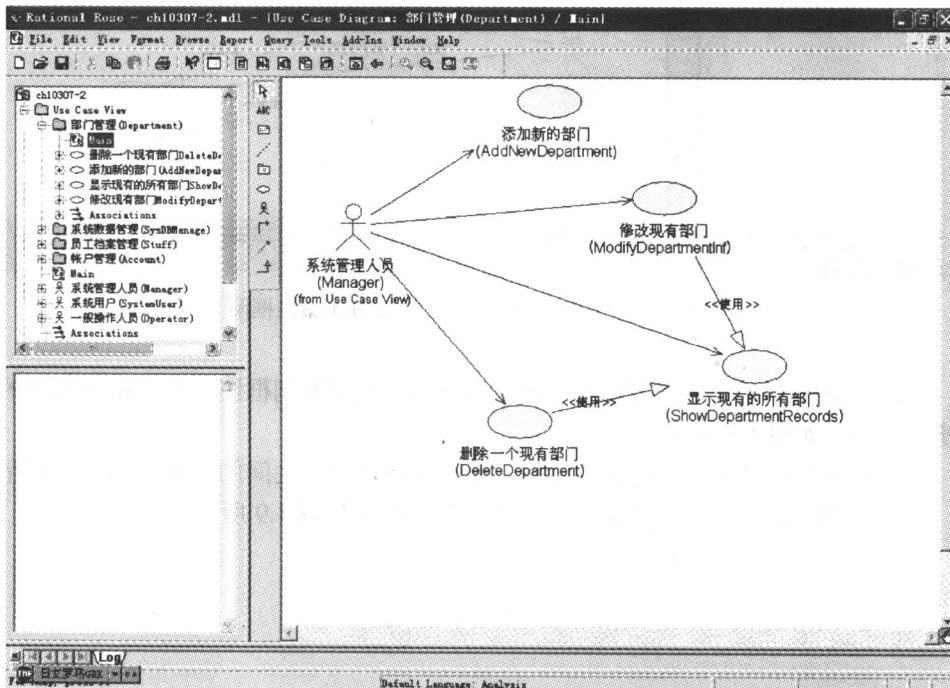


图 1.8 “部门管理”包的 Main 使用案例框图

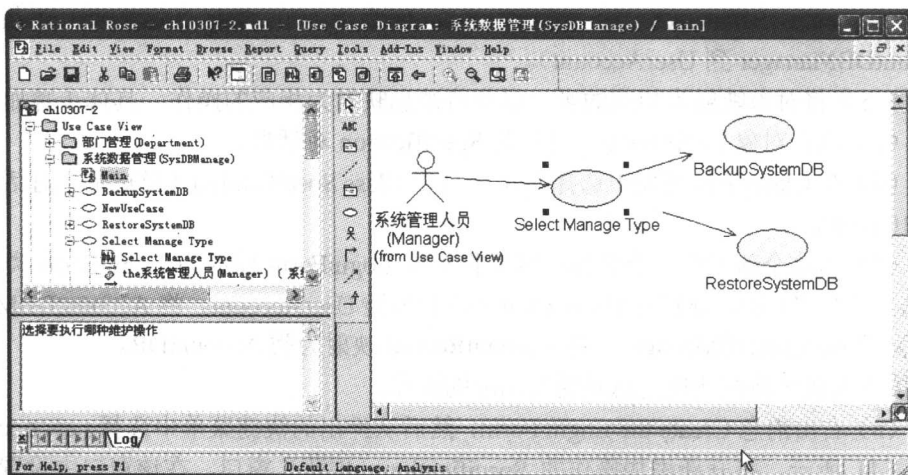


图 1.9 “系统数据管理”包的 Main 使用案例框图

1.1.3 创建工程的逻辑视图

在确定了角色、使用案例以及它们之间的关系后，还应该具体分析每个使用案例中的对象与角色间的交互关系。为此，Rose 向设计人员提供了 Interaction Diagram (Interaction 框图)，Interaction Diagram 又可分为 Sequence 框图和 Collaboration 框图。其中 Sequence 框图是按时间排序的，可以很方便地了解一个使用案例的逻辑流程，而 Collaboration 框图则更清晰地显示了对象之间的关系。下面介绍 Sequence 框图的使用方法。设计人员可以在 Use Case View 中创建 Interaction Diagram，也可以在 Logical View 中创建 Interaction Diagram。

依次选择 Tools | Options，然后选中 Diagram 选项卡。确保其中的 Sequence Numbering 和 Collaboration Numbering 两选项都被选中，这样在随后进行设计时，Rose 会为新添加的对象自动生成序号。下面以 Login 使用案例为例介绍创建 Sequence 框图的方法。

在 Use Case View 中生成 Login 使用案例的 Sequence 框图步骤如下：

(1) 右击右边浏览器中 Use Case View 下“账户管理”包中的 Logic 使用案例 (Sequence Diagram 应该位于相应的使用案例之内)，从弹出的快捷菜单中选择 New | Sequence Diagram，将新添加的 Sequence 框图命名为 LoginDiagram。

(2) 双击 LoginDiagram，打开该框图。将左边浏览器中的角色 SystemUser 拖到右部的 Sequence 框图中。

(3) 使用工具栏中的 Object 工具  向框图中依次添加并命名以下对象：LoginForm、UserAccount、AccountDBManager 和 AccountDB。

(4) 在框图中添加消息，选择工具栏中的 Object Message  图标，从角色 SystemUser 的生命线拖动到 LoginForm 的生命线，并将该消息命名为 Create The Login Form (这里表示首先生成用于登录的登录窗口)。在框图中依次添加消息：Check the account (SystemUser 和 LoginForm 之间) 和 Set the Account Information (LoginForm 和 UserAccount 之间)、Search the Account in Database (Login Form 和 AccountDBManager

以及 AccountDBManager 与 AccountRecord 之间)、Set Operator's Right (从 AccountDBManager 到 UserAccount)。

接下来将对象映射为相应的类, 以及将消息映射为相应的操作, 具体步骤如下:

(5) 双击对象 LoginForm, 打开其 Specification 对话框。

(6) 在 Class 下拉列表中选择 New, 打开 Class Specification (类规范) 对话框, 如图 1.10 所示。

(7) 为这个新类取一个名称, 这里将对象 LoginForm 映射为 LoginForm 类。重复第 (6)、第 (7) 步, 分别将 UserAccount 映射为类 UserAccount, 将 AccountDBManager 映射为类 AccountDBManager, 将 AccountRecord 映射为类 AccountDB。

接下来继续将描述性消息映射为相应的操作。

(8) 右击消息 Create the Login Form, 然后从弹出的快捷菜单中选择 new operation, 如图 1.11 所示。打开新添加操作的 Specification (规范) 窗口, 在该窗口中设置 Name 为 CreateLoginForm, 然后单击 OK 按钮。

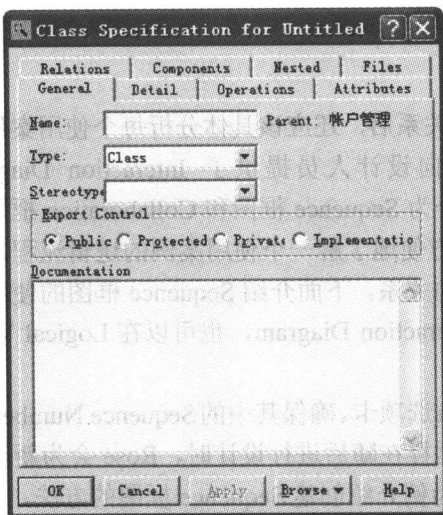


图 1.10 Class Specification 对话框

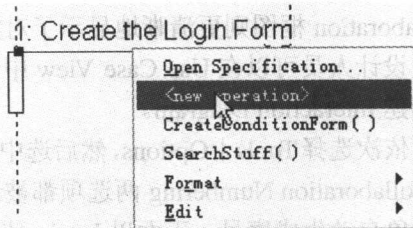


图 1.11 将消息映射为操作 CreateLoginForm

(9) 重复上一步骤, 为其他消息添加相应的操作: 将消息 2 映射为操作 CheckAccount, 消息 3 映射为 SetUserRight, 消息 4 和消息 5 映射为 SearchAccount, 将消息 6 映射为 SetRight。

(10) 完成上述操作后, 依次选择 Tools | Check Model 检查模型的合法性。Login 使用案例的最终 Sequence 框图如图 1.12 所示。

设计人员也可根据需要创建该使用案例的 Collaboration 图, 创建 Collaboration 图的方法与创建 Sequence 框图的方法相同。右击 Login 使用案例, 然后依次选择 new | Collaboration Diagram 创建一个新的 Collaboration 框图, 然后双击该框图并进行设计。

此外, 如果前面已经生成了相应的 Sequence 框图, 也可以直接按下 F5 键, Rose 将自动地生成与该 Sequence 框图相对应的 Collaboration 框图。Login 使用的 Collaboration 框图如图 1.13 所示。

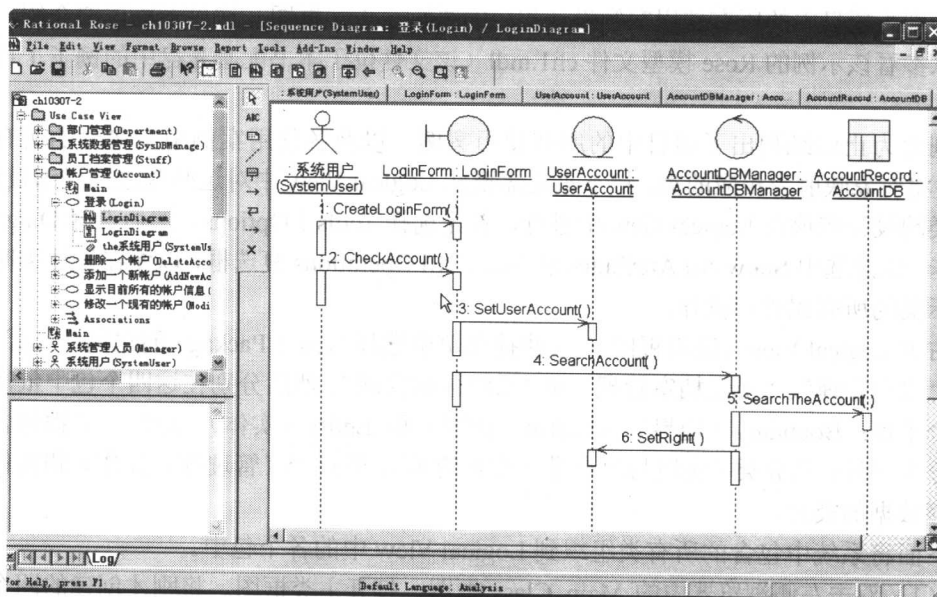


图 1.12 Login 使用案例的 Sequence 框图

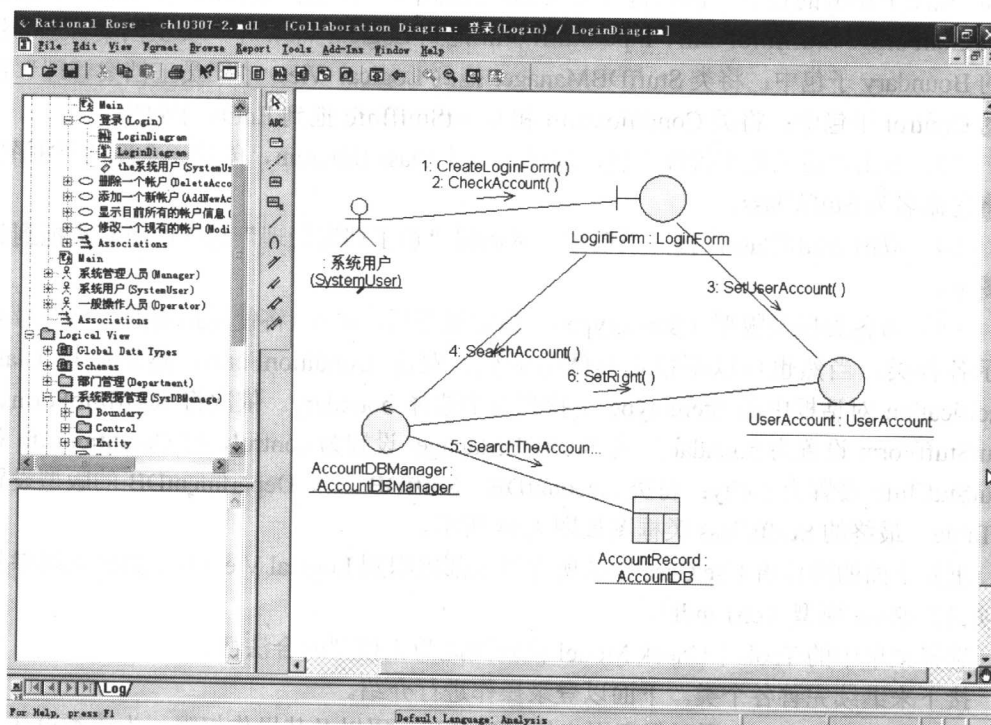


图 1.13 Login 使用案例的 Collaboration 框图