



“十二五”职业教育国家规划教材
经全国职业教育教材审定委员会审定



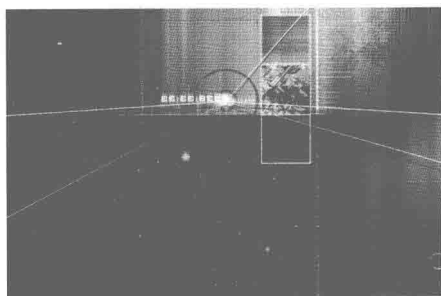
国家级精品资源共享课立项课程配套教材

INDUSTRY AND INFORMATION TECHNOLOGY TRAINING PLANNING MATERIALS
TECHNICAL AND VOCATIONAL EDUCATION

工业和信息化人才培养规划教材

高职高专计算机系列

UML 软件建模技术



Software modeling technology in Unified Modeling Language

国家级精品课程、国家级精品资源共享课立项课程配套教材；按照“需求建模-架构建模-应用建模”这一实际项目的实现过程进行编排；将软件开发工程师应具备的知识、能力和素质有机地融合到项目案例分析与设计

江文 ◎ 主编

唐俊 戴臻 叶晖 王湘渝 ◎ 副主编



人民邮电出版社
POSTS & TELECOM PRESS



精品系列



“十二五”职业教育国家规划教材
经全国职业教育教材审定委员会审定

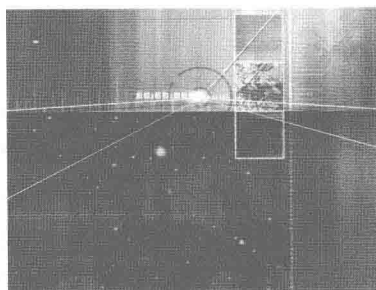


国家级精品资源共享课立项课程配套教材

工业和信息化人才培养规划教材

高职高专计算机系列

UML 软件建模技术



Software modeling technology in
Unified Modeling Language

江文◎主编

唐俊 戴臻 叶晖 王湘渝◎副主编

人民邮电出版社

北京

图书在版编目 (C I P) 数据

UML软件建模技术 / 江文主编. — 北京: 人民邮电出版社, 2015. 1

工业和信息化人才培养规划教材. 高职高专计算机系列

ISBN 978-7-115-35208-8

I. ①U… II. ①江… III. ①面向对象语言—程序设计—高等职业教育—教材 IV. ①TP312

中国版本图书馆CIP数据核字(2014)第063906号

内 容 提 要

本书以实用、够用为原则,介绍了软件建模技术的基本概念。全书内容由浅入深,逐步展开,并配有大量的案例和课堂练习,力图使初学者更容易理解。

本书从最基本的内容讲起,详细介绍了UML语言,并着重论述了如何使用UML对应用系统进行建模。同时,为配合知识点的讲述,将一个实际项目作为案例与所讲述的内容融合在一起,力图做到从应用中来到应用中去,例如用类图和交互图来描述诚信公司的诚信管理论坛系统中的静态和动态行为。本书是一本以知识为导向,以实际应用为目标的讲解软件建模技术的教材。

本书可作为职业院校软件专业课教材,也可供相关从业人员和技术人员参考。

-
- ◆ 主 编 江 文
 - 副 主 编 唐 俊 戴 臻 叶 晖 王 湘 渝
 - 责任编辑 王 威
 - 责任印制 杨林杰
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
 - 邮编 100164 电子邮件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 北京鑫正大印刷有限公司印刷
 - ◆ 开本: 787×1092 1/16
 - 印张: 14 2015年1月第1版
 - 字数: 370千字 2015年1月北京第1次印刷
-

定价: 36.00 元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

反盗版热线: (010)81055315

前 言

软件建模是通过建立一套模型在待开发软件系统需求与实现之间架起一座桥梁，如同建造一栋大楼之前需要绘制设计施工图一样，这样软件设计师与工程师就能按照所建立的模型开发与改进软件系统。UML（Unified Modeling Language，统一建模语言）是一种通用的可视化建模语言，用于对软件系统进行描述、可视化处理、构造和建立系统的工作文档。它记录了与被设计系统有关的决策和分析，可用于对系统进行分析、设计、浏览、配置、维护以及控制。UML 具有灵活、表达能力强的特点，是目前主流的软件建模语言。

软件设计技术类课程目前已成为高职院校计算机类专业教学中的重要课程，是计算机类专业学生必须掌握的专业技能之一。根据对软件企业的软件设计师、程序员、测试员等职业岗位的知识、技能和素质分析，结合高职学生的认知规律和专业技能的形成规律，为使学生熟练掌握软件设计的基本理论和技术，不少高职院校开始将“软件建模技术”作为重要的专业必修或选修课程开设。

本书是一本为高职院校“软件建模技术”课程“讲练一体”教学量身定做的教材，选用行业主流的“Enterprise Architect”作为建模工具平台，主要介绍使用例图、类图、活动图、时序图、组件图和部署图对应用系统进行需求分析、组织架构和应用建模等方面的知识。课程的学习情境是设计与开发诚信管理论坛系统。本书对“诚信管理论坛系统”项目案例进行剖析与分解，并对课程知识点进行重构和组合，模拟相应的学习情境，不仅帮助学生掌握软件项目开发中的需求分析与设计等方面的专业知识与技能，还能够全面培养学生的综合素质，提高其收集资料的能力、检查判断的能力、合理使用工具的能力、组织协调能力、语言表达能力、责任心与职业道德、自我保护能力、应变能力，同时，通过工学结合的学习掌握工作岗位需要的各项技能和相关专业知识。

本书在对诚信管理论坛系统和在线聊天系统进行解析的基础之上，将软件开发工程师应具备的知识、能力和素质训练有机地融合到项目的分析与设计中，形成 4 个理实一体化的教学单元。课程考核采取项目开发与过程考核相结合的方式。

在通过了“十二五”职业教育国家规划教材选题立项之后，根据《教育部关于“十二五”职业教育教材建设的若干意见》对本书进行了修订，邀请行业、企业专家和一线课程负责人一起，从人才培养目标、专业方案等顶层设计做起，明确了软件建模课程标准；强化了教材的沟通与衔接，力求在中高职衔接上平滑过渡；根据岗位技能要求，引入了企业真实案例，增加了“项目实战”模块；重点建设了课程配套资源库，新增了配套光盘，建设了课程教学网站，通过“微课”等立体化的教学手段来支撑课堂教学。力求达到“十二五”职业教育国家规划教材的要求，提高高职院校软件建模课程教学质量。

我们对本书的体系结构做了精心的设计，按照“需求建模—架构建模—应用建模”这一

实际的项目实现过程进行编排,力求将需求分析、设计和实现这三者有机地结合在一起,体现软件开发实现的全过程。在内容编写方面,本书难点分散,循序渐进;在文字叙述方面,本书用词浅显易懂,重点突出;在实例选取方面,案例实用性强,针对性强。

各教学单元设计如下表所示。

各教学单元及任务列表

教学单元	教学任务	参考学时	
项目一 软件建模基础知识	1.1 软件建模概述	1	3
	1.2 UML 与 Enterprise Architect 建模工具	2	
项目二 需求建模	2.1 用例图	2	8
	2.2 诚信管理论坛系统需求分析	2	
	2.3 技能提升——在线聊天系统需求分析	1	
	2.4 活动图	2	
	2.5 技能提升——在线聊天系统需求动态建模	1	
项目三 架构建模	3.1 状态图	4	14
	3.2 类	2	
	3.3 类图与类的关系	4	
	3.4 交互图	4	
项目四 应用建模	4.1 对象图和包	2	7
	4.2 组件图和部署图	2	
	4.3 正向工程与逆向工程	3	

本书每个项目都附有作业,可以帮助学生进一步巩固基础知识。本书配备了 PPT 课件、源代码、课程标准等丰富的教学资源,任课教师可到人民邮电出版社教学服务与资源网(www.ptpedu.com.cn)免费下载使用。本书由湖南科技职业学院的江文担任主编,项目一由湖南科技职业学院江文编写,项目二由戴臻编写,项目三由唐俊与王湘渝编写,项目四由长沙学院叶晖与湖南科技职业学院邓军编写。

由于编者水平有限,书中难免存在不足之处,敬请广大读者批评指正。

编者

2014年2月

目 录 CONTENTS

项目一 软件建模和软件工程 1

1.1 软件建模概述	1	1.2.5 Rational 统一过程	15
1.1.1 软件建模概述	2	1.3 UML 基本组成	19
1.1.2 UML 简介	3	1.3.1 UML 事物	19
1.1.3 建模工具 Enterprise Architect	6	1.3.2 UML 关系	22
1.2 软件工程与 Rational 统一过程	10	1.3.3 UML 图	24
1.2.1 软件	10	1.4 扩展阅读——Rational Rose 工具	
1.2.2 软件危机	12	简介	27
1.2.3 软件工程	12	1.4.1 建模工具 Rational Rose 简介	28
1.2.4 面向对象软件工程方法	14		

项目二 需求建模 35

2.1 用例图	36	2.4.1 活动图的基本概念	58
2.1.1 参与者	37	2.5 技能提升——在线聊天系统需求动	
2.1.2 用例	38	态建模	74
2.1.3 用例图	41	2.6 扩展阅读——面向对象需求分析	
2.1.4 用例与事件流	42	方法	75
2.1.5 用例之间的关系	43	2.6.1 面向对象的概念与特征	75
2.2 诚信管理论坛系统需求分析	48	2.6.2 面向对象软件开发的分析模型	78
2.3 技能提升——在线聊天系统需求		2.6.3 基于 UML 的软件开发过程	80
分析	55	2.6.4 面向对象的需求分析	82
2.4 活动图	58		

项目三 架构建模 89

3.1 状态图	90	3.2.1 类	104
3.1.1 事件	90	3.2.2 类成员的可见性	106
3.1.2 状态	93	3.2.3 类的类型和类的寻找	107
3.1.3 转换	94	3.2.4 技能提升——寻找在线聊天系统	
3.1.4 状态图	96	中的类	116
3.1.5 技能提升——在线聊天系统服务器		3.3 类的关系	118
运行状态建模	101	3.3.1 依赖	119
3.2 类	103	3.3.2 泛化	121

3.3.3	实现	122	3.4.3	技能提升——在线聊天系统类的 动态建模	140
3.3.4	关联	123	3.5	扩展阅读——面向对象设计	141
3.3.5	技能提升——在线聊天系统类的 关系建模	130	3.5.1	面向对象设计的任务	142
3.4	交互图	132	3.5.2	面向对象设计的准则	143
3.4.1	顺序图	132	3.5.3	启发性规则	145
3.4.2	协作图	134	3.5.4	面向对象设计过程	146
			3.5.5	面向对象设计实例	153

项目四 应用建模 162

4.1	对象图和包	163	4.2.3	技能提升——在线聊天系统应用 建模	185
4.1.1	对象图	163	4.3	正向工程与逆向工程	187
4.1.2	包	165	4.3.1	正向工程	188
4.1.3	技能提升——在线聊天系统对 象图	172	4.3.2	逆向工程	192
4.2	组件图和部署图	174	4.3.3	技能提升——在线聊天系统正向 工程	196
4.2.1	组件图	174			
4.2.2	部署图	180			

软件建模技术理论考核试卷（一） 200

软件建模技术理论考核（二） 205

软件建模技术理论考核试卷（三） 219

软件建模技术习题（四） 214

参考文献 218

PART 1 项目一

软件建模和软件工程

本项目目标

模型 (Model) 是对现实世界的简化, 软件建模则是对业务系统软件的抽象描述。通常在软件设计与分析中使用 UML 语言来建模。UML (Unified Modeling Language, 统一建模语言) 是一种可视化的建模语言, 主要应用于软件工程领域。本项目主要是一些基本概念的描述, 因此非常重要。通过本项目的学习, 我们将理解软件建模和软件工程的主要概念, 为后续的学习打好基础。本项目的学习目标如下。

- 理解建模的概念。
- 理解软件工程的基本概念。
- 理解软件建模基本概念以及建模语言的组织结构。

1.1 软件建模概述



内容提要

模型 (Model) 是对现实世界的简化, 软件建模则是对业务系统软件的抽象描述。通常在软件设计与分析中使用 UML 语言来建模。UML 是一种可视化的建模语言, 它可以用来创建各种不同类型的模型。本节将首先讲述建模的概念, 然后引出建模语言——UML。在 UML 这一小节中主要介绍了 UML 的历史、UML 的基本概念。另外, 必须使用一种工具来帮助我们实现 UML 建模, 因此在本节的最后介绍了 UML 建模工具 Enterprise Architect。本节主要内容如下。

- 建模概述。
- UML 简述。
- Enterprise Architect 介绍。

1.1.1 软件建模概述

1. 什么是模型

什么是模型？在回答这个问题之前，我们先来回忆一下生活中常见的一些图表、文字：介绍天气情况的气象图；指示交通情况的交通地图；说明泡沫式灭火器如何打开的过程描述图……所有这些，都是我们身边事物的模型。那么，模型是什么呢？简单地说，模型是对现实的简化。它是现实事物的一种微缩表示，或是一种用于生产某事物的模式，也可以是一种设计或类型，还可以是一个待模仿或仿真的样例。一个好的模型应包括那些有广泛影响的主要元素，忽略那些与抽象水平不相关的次要元素，如：在对房屋进行建模中应包括房屋材料、构造结构等主要元素，忽略那些房屋中应摆放什么样的家具等次要元素。每个系统都可以从不同的方面用不同的模型来描述，如：在建筑、机械设计中就会有用正视、侧视等视图来表示事物的效果，这正如中国一句古诗所说“横看成岭侧成峰”。另外，模型不一定是可视化的，模型也可以用文字来描述，比如用文字描述车间里一个产品的生产流程，但是可视化模型可以更准确地展示模型所代表的含义。

2. 建模的目的和原则

我们为什么要建模？其主要理由是通过建模能够更好地理解我们正在开发的系统。在开发系统的时候，建模可以帮助我们沟通设计思想，理解业务内容，以及处理流程，澄清复杂的问题和场景，确保所设计的系统在实现之前能更符合用户需求。按照这种模式来思考，我们会在没有规划之前就开工建

造一栋大厦吗？建造狗窝也许不需要详细的规划和设计，但想要建造好一栋大厦，不把大厦建成像狗窝一样，就需要事先仔细设计一番了。当我们使用一套好的设计图纸，并严格依照图纸施工，所建造的大厦才能经得起时间的检验。而且，越是复杂、庞大的系统，就越需要通过事先的建模来设计与规划。因为人们对复杂问题的理解能力是非常有限的，只有通过建模来帮助人们理解复杂的问题，每次只研究复杂系统的一个方面，即先把待解决的难题分解成一系列的小问题，解决了这些小问题也就解决了这个难题。

什么样的模型才是合乎要求的呢？如果所建的模型对我们的工作没有多大的帮助，或者对我们的工作反而有误导作用，这样的模型建立出来没有什么作用。因此，建模时要有明确的目的性，不要为了建模而建模，也不要事事都建模。当我们专注于建模并希望它产生效力时，就需要先分析从建模中是否能获得收益，或者说值不值得建模。事实上我们发现，项目越简单，建模发挥的功效就越小。一般来说，通过建模，要达到以下4个目的。

- (1) 模型帮助我们按照实际情况对系统进行可视化。
- (2) 模型允许我们详细说明系统。
- (3) 模型给出了一个指导我们构造系统的模板。
- (4) 模型对我们做出的决策进行模板化。

建模并不是一个刚刚冒出来的新鲜事物，事实上，在各种传统的工程科学领域都有丰富的

建模历史，这些经验形成了建模的一些基本原则，有以下几点。

(1) 要仔细地选择模型。

创建什么样的模型对解决问题有着重大的影响。正确的模型将清晰地阐明所要开发的系统，而错误的或是有偏差的模型将误导我们把精力放在不相关的问题上。

(2) 每一种模型可以在不同的精度级别上表示所要开发的系统。

举个例子来说，修建一栋大厦，有时我们需要使用 3D 软件制作大厦的整体视觉图，供投资者参考；有时需要制作一份详细的电气施工图，供大厦的施工员铺设电线、光纤。由此可以看到，在项目开发中，不同的角色其视角不同，对模型的侧重方面和详细程度则有不同的要求，如：系统的使用者主要考虑“能做什么”的问题，而系统的开发人员则更多考虑的是“如何做”的问题，这两类人就需要从不同角度以不同的精度级别对系统进行可视化建模。

(3) 模型要与实际相联系。

一个模型如果脱离了实际，显然这不是一个好的模型。因此，我们要注意一点，虽然模型对现实进行了简化，但不能简化掉任何重要细节，也不能改变或歪曲任何重要细节。

(4) 对一个重要的系统用一组几乎独立的模型去处理。

对于复杂的或者是重要的系统，只用单个模型来描述往往是不够充分的，这时需要用多种模型对系统分别进行研究和描述，以加深我们对系统的理解。

3. 使用 UML 建模

前面讲述了在工程领域里建模的重要性，那么，我们该如何建模呢？先来看看其他的领域：在音乐领域，有五线谱，供作曲家和演奏家交流；在数学领域，有各种各样的数学公式和表示方法，供数学家、教师、学生交流学习。同样，在工程领域，也有一种可以提供给工程设计人员使用的公共语言：UML。UML 的中文意思是统一建模语言（Unified Modeling Language），它是一种通用的可视化建模语言。UML 具有灵活，表达能力强的特点。有了建模语言，就方便我们对各种工程进行描述、分析和交流。针对本书而言，论述的是如何使用 UML 在软件工程方面建模，所以下面将简单的介绍一下有关 UML 的知识，关于软件工程，将在下一节中进行论述。

1.1.2 UML 简介

1. UML 历史

要了解 UML，就有必要从它的源头开始。20 世纪 80 年代末，出现了许多面向对象的软件建模技术，这些技术是由不同的专家学者发明的，也使用了不同的建模技术和模型表示法。但是，采用面向对象分析与设计方法的用户，并不一定了解各种建模技术、语言之间的差异，因此很难把握为其所开发的应用系统选择合适的建模语言。直到 20 世纪 90 年代中期，有 3 种面向对象建模方法逐渐占据了统治地位，分别是 Jim Rumbaugh 的对象建模技术（OMT）、Ivar Jacobson 的面向对象软件工程方法（OOSE）和 Grady Booch 的 Booch 方法。1994 年，Rational 公司聘请了 Rumbaugh 参加 Booch 的工作。两人合并了 OMT 和 Booch 方法中的概念与方法，

并于1995年提出了第一个建议方案。同年，Jacobson也加入了Rational公司，三位最优秀的面向对象方法学的创始人终于聚在了一起，他们共同的研究成果就是统一建模语言（UML）。1997年，Rational公司正式将UML 1.0版作为标准草案提交给独立标准化组织OMG（Object Management Group，对象管理组织）并获得通过。此后，OMG承担了进一步完善UML标准的工作，并先后推出了UML的多个版本。

有了若干年使用UML的经验之后，OMG提出了升级UML的建议方案，以修正使用中发现的问题，并扩充一部分应用领域中所需的额外功能。建议方案自2000年11月开始起草，至2003年7月完成。之后，UML 2.0规范被全体OMG会员采纳并正式发布。总的来说，UML 2.0和UML 1.0大部分是相同的，尤其是核心特征。

2. UML 简述

统一建模语言UML（Unified Modeling Language）是一种通用的可视化建模语言，用于对软件系统进行描述、可视化处理、构造和建立系统的工作文档。它记录了与被设计系统有关的决策和分析，可用于对系统的分析、设计、浏览、配置、维护以及控制。UML包括语义概念、表示法和指导规范，提供了静态、动态、系统环境和组织结构等类型的模型。UML能够捕捉系统静态结构和动态行为的信息。静态结构定义了系统中重要对象的属性和操作，以及这些对象之间的关系。动态行为定义了对象随时间变化的历史和对对象为完成目标而进行的相互通信。UML能从不同的角度对系统进行建模，因此可以全方位的帮助用户了解和分析系统。

UML体系由3个部分组成：UML基本构造块、UML规则和UML公共机制。只有当我们掌握了这些内容，才能够读懂UML模型，并且能根据应用系统要求构建相应的系统模型。

（1）UML 基本构造块。

UML有3种基本构造块，分别是事物、关系和图。事物包括结构事物、行为事物、分组事物、注释事物4种。关系包括依赖关系、关联关系、泛化关系、实现关系4种。图包括类图、对象图、用例图、顺序图、协作图、状态图、活动图、组件图、部署图9种。关于事物、关系和图这3种构造块及其各自的组成部分，本书稍后将会详细地描述。

（2）UML 规则。

一个结构良好的模型在语义上应该是前后一致的，并且与所有的相关模型协调一致。因此，我们不能简单地把UML构造块随机摆放在一起堆砌成一个模型。UML通过定义一套规则来告诉我们如何使用UML构造块搭建出一个结构良好的模型。UML有用于描述如下事物的语义规则：“命名”、“范围”、“可见性”、“完整性”、“执行”等。

（3）UML 公共机制。

公共机制是指达到特定目标的公共方法，在UML中有多种贯穿整个语言的公共机制，主要包括：规格说明、修饰、通用划分和扩展机制。UML规格说明提供了对构造块的语法和语义上的描述，如：对类的注释。修饰是对UML元素特性进行描述的符号，也就是说可以通过这些修饰进一步表达元素信息，如图1.1.1中的类图中，类中的属性与方法之前加上各种表示其可访问性的修饰，其中“+”表示该属性或方法为公有类型。通用划分是对UML元素按其功能与作用进行划分，目前UML包括两组公共分类——类与对象，类表示概念的抽象，而对象表示

具体的事物；接口与实现，接口是用来定义契约，实现则是对契约的具体实现。扩展机制是对 UML 元素的扩展，包括约束、构造类型和标记值。约束可以扩展 UML 元素的语义，允许增加新的规则或修改现的规则；构造类型扩展了 UML 的词汇，它允许创建新的构造块；标记值则是扩展 UML 构造块的特性，允许创建新的特殊信息来扩展事件的规格说明。

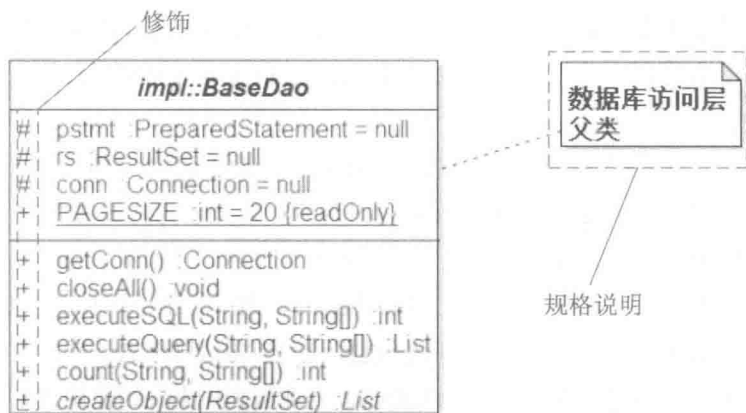


图 1.1.1

在应用中，我们不必了解或掌握 UML 中的每项特征，就像我们不需要知道或使用大型项目中的每项功能一样。通常被广泛使用的只有核心概念这一小部分，其他的特征可以逐步学习，在需要的时候再使用。UML 合并了许多面向对象方法中被普遍接受的概念，对每一种概念，UML 都给出了清晰的定义、表示法和相关术语。这样，一个开发者用 UML 绘制模型，而另外一个开发者可以无歧义地解释这个模型。

UML 本质上不是一门编程语言。但是，人们可使用代码生成器将 UML 模型转换为多种程序语言的程序代码，或使用反向生成工具将代码还原成 UML 模型。UML 也不是一种用于定理证明的高度形式化的语言。UML 本质上只是一种通用的建模语言。

3. UML 的应用领域

UML 的目标是以面向对象的方式来描述任何类型的系统，其中最常用的是建立软件系统模型。UML 的设计初衷是支持面向对象系统建模，以及基于构件的开发。但是，在 UML 的设计中也考虑了其他需求，今天，通过使用 UML 内置的扩展和用户定制能力，UML 同样也可以用来描述非软件领域的系统，如机械系统、企业机构或业务过程，以及处理复杂数据的信息系统、具有实时要求的工业系统或工业过程等。总之，UML 是一个通用的标准建模语言，可以对任何具有静态结构和动态行为的系统进行建模。

UML 适用于系统开发过程中从需求规格描述到系统完成测试后的不同阶段。例如，在需求分析阶段，可以用用例来描述客户的需求；在设计阶段，可以用 UML 动态模型来描述对象与对象之间的关系；在测试阶段，UML 模型还可以作为测试的依据。

1.1.3 建模工具 Enterprise Architect

Enterprise Architect (注: 以下简称为 EA) 是由澳大利亚 Sparx Systems 公司设计开发的一套软件辅助开发平台。EA 是基于 UML 2.4 标准设计开发的, 具有从业务需求到软件设计, 直至部署的软件开发全过程的设计与管理能力。该平台还可以根据所创建的 UML 模型生成 Java、C#、C++ 等 10 余种源程序代码。目前 EA 在全球拥有 30 多万注册用户, 是当前主流的软件建模与管理工具。

Sparx Systems 公司根据用户不同需求发布了企业版、专业版和桌面版 3 种 EA 版本。企业版是针对大型开发团队的, 它提供了模型审核、版本控制、以角色为基础的安全设置、思维导图、选择 DBMS 库等功能。专业版是适用于工作组级的开发平台, 它为专家和开发人员提供了功能强大的建模与可视化工具, 支模型正逆向工程、数据库工程、思维导图和需求变化跟踪等功能。桌面版是为单个的分析与开发人员定制的业务过程建模与测试管理的工具, 主要提供了业务建模、思维导图与测试管理等功能。本书将以 EA 9.0 企业版为例来介绍使用该工具进行系统建模的方法。

1. 启动 EA

启动 EA 9.0 后, 进入到主界面, 首先弹出如图 1.1.2 所示的对话框, 在该对话框中可以创建、打开项目工程。



图 1.1.2

进入到 EA 工作界面后, 该界面是由标题栏、菜单栏、工具栏、工作区和状态栏组成, 如图 1.1.3 所示。工作区的右侧是树形视图和属性区, 每选中树形视图的某个对象, 文档区就会显示其对应的文档名称与内容; 左侧是编辑区, 在该区中可以打开模型中的任意一张图, 并可利用工具栏对图进行修改。

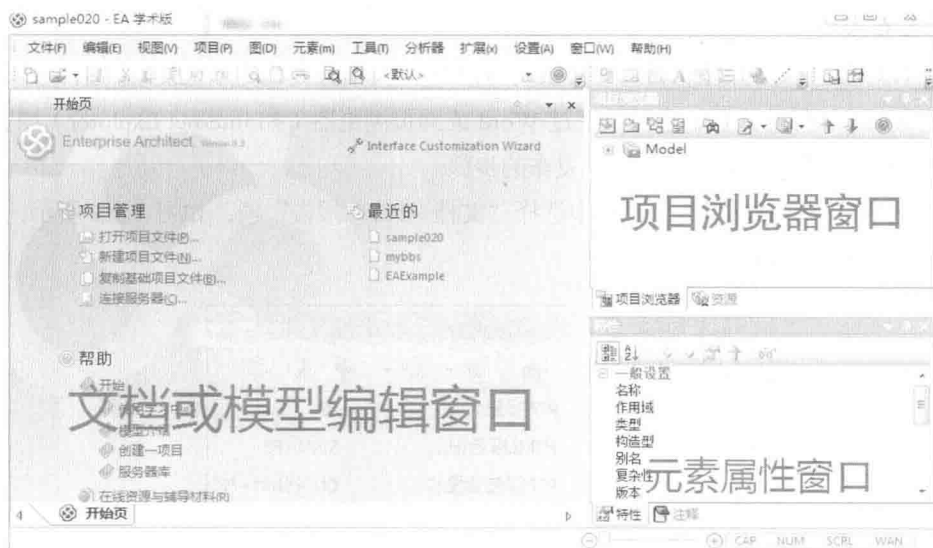


图 1.1.3

2. 创建模型

使用 EA 创建软件模型工程有两种方法：一种是在启动 EA 时使用启动对话框来创建项目工程；另一种方法则是采用选择主菜单“文件→新建项目...”项的方法来创建。在创建项目工程时，首先要求输入新建的项目工程文件，EA 的项目工程文件是以“.eap”为后缀名的，接着系统要求选择创建模型的类型。目前有“Basic UML 2 Technology（基本的 UML 模型）”、“Core Extensions（核心扩展）”、“Entity Relationship Diagram（实体关系图）”等 5 种类型，如图 1.1.4 所示。如果所创建的项目工程只是用于软件建模，则可以选择“Basic UML 2 Technology”，并在该技术类型中选择建模时需要使用到的具体的模型，如：用例图、域模型、类、组件图和部署图。

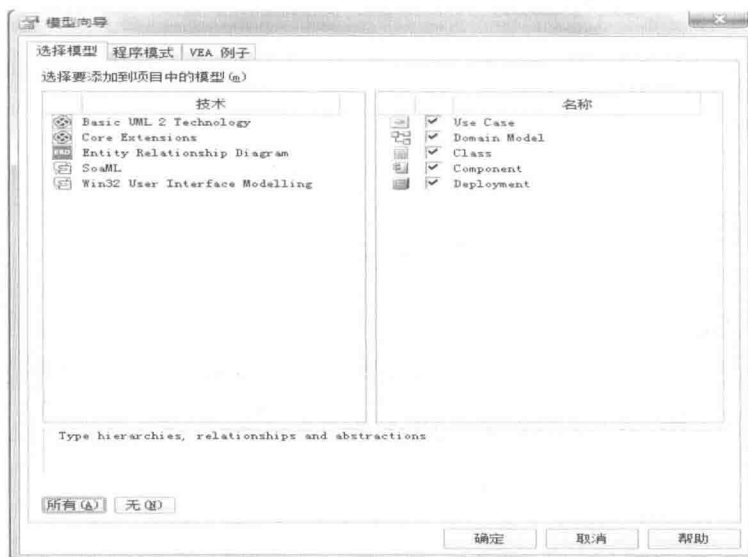


图 1.1.4

3. 发布模型

EA 可以把建立好的模型以 RTF 文档或 HTML 网页格式文档的形式发布, 这样可以让其他即使没有装 EA 软件的人员, 也可以通过 Word 或网页浏览器 (如 Internet Explorer) 浏览模型。下面以发布 RTF 格式文档为例来讲解发布的步骤。

(1) 从项目浏览器窗口的工具栏中选择“文档→RTF 报告”项, 如图 1.1.5 所示, 选该菜单之后将弹出如图 1.1.6 所示的对话框。



图 1.1.5

- (2) 在弹出的对话框中输入发布后文档的文件名以及文档生成模板, 如图 1.1.6 所示。
 (3) 设置完毕, 单击“运行”按钮, 即可进行模型发布。



图 1.1.6

4. 设置全局属性

从菜单栏选择“工具→选项”项, 将弹出如图 1.1.7 所示的对话框。在该对话框里可以设置一些全局属性, 如字体、颜色、正向工程源代码模板设置等。

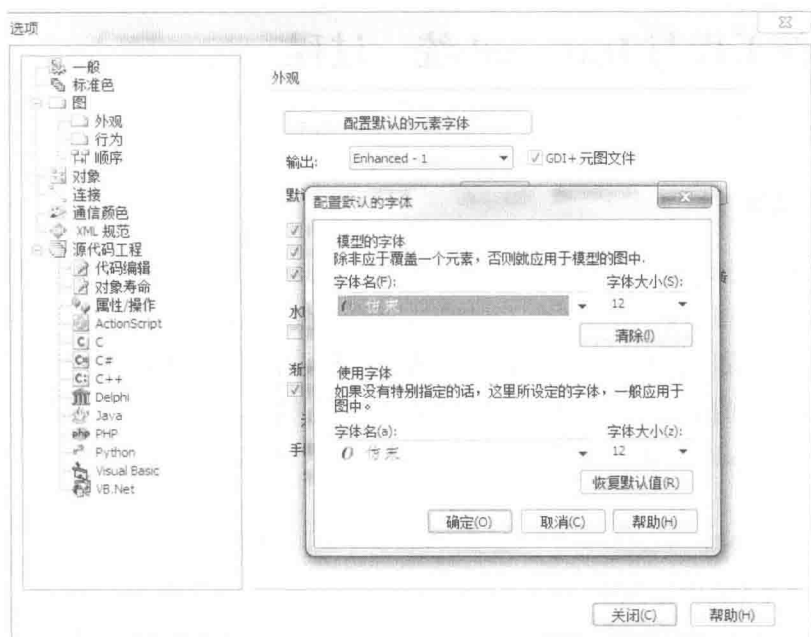


图 1.1.7



小结

本节我们主要学习了以下知识。

1. 建模

(1) 模型的概念。

模型是对现实的简化。它可以是一个对象的微缩表示, 或是一种用于生产某事物的模式, 也可以是一种设计或一个类型, 还可以是一个待模仿或仿真的样例。

(2) 建模的目的和原则。

建模是为了能够更好地理解我们正在开发的系统。

建模的原则包括: ①要仔细地选择模型; ②每一种模型可以在不同的精度级别上表示所要开发的系统; ③模型要与现实相联系; ④对一个重要的系统用一组几乎独立的模型去处理。

2. UML 概述

统一建模语言是一种通用的可视化建模语言, 用于对软件进行描述、可视化处理、构造和建立软件系统的工作文档。UML 的组成部分包括: UML 基本构造块、UML 规则和 UML 公共机制。

3. UML 建模工具 EA 的使用方法

1.2 软件工程与 Rational 统一过程



内容提要

软件是包括程序、数据及其相关文档的完整集合，其开发过程至今尚未摆脱手工艺的开发方式，随着软件复杂度和开发难度的日益增加，逐渐形成了所谓的软件危机。为了解决软件危机，计算机科学家提出了软件工程的观念——使用工程化的原则和方法组织软件开发工作。本节重点介绍了软件的生存期和几种典型的软件生存期模型，最后介绍了 Rational 统一过程 (Rational Unified Process, RUP)。本节主要内容如下：

- 软件
- 软件生命周期
- 软件生存期模型
- RUP

1.2.1 软件

在了解软件工程之前，我们首先需要知道什么是软件。一些人认为，软件就是程序，就是代码，这是一种片面的理解。软件一词最早是 20 世纪 60 年代初从海外传来的，当时很多人都无法给出确切的含义。现在普遍认可的定义是：软件是计算机系统中与硬件相互依存的另一部分，它是包括程序、数据及其相关文档的完整集合。其中，程序是按照事先设计的功能和性能要求执行的指令序列，数据是使得程序能够适当地操作信息的数据结构，文档是描述程序的开发、操作和维护的文字或图形资料。

要理解软件的概念，需要先了解软件的特征，这样就能够理解软件与其他事物之间的区别了。软件是一种逻辑实体，而不是具体的物理实体，因此，它具有与硬件完全不同的特征。

1. 软件是被设计开发的，而不是被制造的

虽然软件开发和硬件制造之间有一些相似之处，例如都可以通过良好的设计得到高质量的产品，但两类活动存在本质上的不同。软件的开发过程中没有明显的制造过程，因此，硬件在制造过程中可以进行质量控制，而这种情况对软件而言几乎是不存在的。如果要对软件进行质量控制，就必须在软件开发和测试方面下工夫。另外，就软件集中开发的特点而言，软件项目也不能像制造类项目那样管理。

2. 软件不会“磨损”，但会“退化”

硬件在运行和使用期间，会有机器磨损、老化等问题。如：硬件在其生命初期有较高的故障率，这些故障主要是设计或制造的缺陷；在中期，随着缺陷的修正，故障率会维持在一个较低的水平上；到了后期，故障率又提升了，这是因为硬件已经进入了“磨损期”。而软件的情况