**第2章 面向对象技术和建模基础**

2.1面向对象的基本概念

面向对象的程序设计思想是将数据以及对于这些数据的操作，封装在了一个单独的数据结构中。这种模式更近似于现实世界，在这里，所有的对象都同时拥有属性及与这些属性相关的行为。

面向对象程序设计是一种新兴的程序设计方法，或者是一种新的程序设计规范，它使用对象、类、继承、封装、消息等基本概念来进行程序的设计。从现实世界中客观存在的事物（即对象）出发来构造软件系统，并且在系统构造中尽可能运用人类的自然思维方式。

2.1.1面向对象方法

面向对象方法的出发点和基本原则是尽可能模拟人类习惯的思考问题的方式，使软件开发的方法与过程尽可能接近人类认识世界、解决问题的方法与过程。由于客观世界的问题都是由客观世界中的实体及实体相互间的关系构成的，因此把客观世界中的实体抽象为对象。

面向对象所带来的好处是程序的稳定性与可修改性（由于把客观世界分解成一个一个的对象，并且把数据和操作都封装在对象的内部）、可重用性（通过面向对象技术，不仅可以重用代码，而且可以重用需求分析、设计、用户界面等）。

面向对象方法具有以下几个要点。

（1）认为客观世界是由各种对象组成的，任何事物都是对象，复杂的对象可以由比较简单的对象以某种方式组合而成。

（2）把所有对象都划分成各种对象类，每个对象类都定义了一组数据和一组方法，数据用于表示对象的静态属性，是对象的状态信息。

（3）按照子类与父类的关系，把若干个对象类组成一个层次结构的系统。

（4）对象彼此之间仅能通过传递消息进行联系。

2.1.2对象

对象（Object）是面向对象的基本构造单元，是系统中用来描述客观事物的一个实体。 一个对象由一组属性和对属性进行操作的一组方法组成。

对象不仅能表示具体的实体，也能表示抽象的规则、计划或事件，主要有如下几种对象类型。

1. 有形的实体
2. 作用
3. 事件
4. 性能说明

对象具有如下特征。

1. 模块性
2. 继承
3. 动态连接性

2.1.3类

一个类定义了一组大体上相似的对象。一个类所包含的方法和数据描述一组对象的共同行为和属性。类是在对象之上的抽象，有了类以后，对象则是类的具体化，是类的实例。类可以有子类和父类，形成层次结构。

类是对事物的抽象，它不是个体对象，而是描述一些对象的完整集合。

把一组对象的共同特性加以抽象并储存在一个类中，是面向对象技术最重要的一点；是否建立了一个丰富的类库，是衡量一个面向对象程序设计语言成熟与否的重要标志。

类是静态的，类的语义和类之间的关系在程序执行前就已经定义好了，而对象是动态的，对象是在程序执行时被创建和删除的。

2.1.4封装

封装就是把一个对象的方法和属性组合成一个独立的单位，并尽可能隐蔽对象的属性、方法和实现细节的过程，仅仅将接口进行对外公开。

在访问类的时候，根据其封装的特点，对外访问时提供了以下4种访问控制级别。

（1） public：公有访问。最高一级的访问，所有的类都可以访问。

（2） protected：受保护的。只有同一个包中的类或者子类可以进行公开访问。

（3） private：私有访问。最低一级的访问，只能在对象的内部访问，不对外公开。

（4） default：默认的。属于当前目录（包）下的类都可以访问。

因此，根据类的封装性，在对属性和方法进行访问时，就需要知道其访问的控制级别，否则是不能使用的。实质上，封装包含两层含义：一层是把对象的全部属性和方法结合在一起，形成一个不可分割的独立单位，对象的属性（除了公有访问的属性）只能由这个对象的方法来存取；二层是极大可能地隐蔽了对象的内部实现细节，与外部的联系只能通过外部接口来实现。

封装的最大优点如下。

1. 方便了使用者对类和对象的操作，并降低了使用者错误修改其属性的机率。
2. 体现了系统之间的松散耦合关系并提高了系统的独立性。
3. 提高了程序的复用性。
4. 针对大型的开发系统，降低了开发风险。如果整个系统开发失败，一些相对独立的子系统仍然存在可用价值。

2.1.5继承

继承是一种一般类与特殊类的层次模型。继承性是指特殊类的对象具有其一般类的属性和方法，在其之上又增加了自己的特殊属性和方法。继承意味着在特殊类中不用重新定义在一般类中已经定义过的属性和方法，特殊类可以自动地、隐含地拥有其一般类的属性与方法。继承体现了类之间代码的重用性特点，提供了一种明确表达共性的方法。

继承是一种一般类与特殊类的层次模型。继承性是指特殊类的对象具有其一般类的属性和方法，在其之上又增加了自己的特殊属性和方法。继承意味着在特殊类中不用重新定义在一般类中已经定义过的属性和方法，特殊类可以自动地、隐含地拥有其一般类的属性与方法。继承体现了类之间代码的重用性特点，提供了一种明确表达共性的方法。

在继承中，需要明确这样两个概念：子类和父类。

子类：指的是通过继承创建的新类，称为“子类"或者“派生类

父类：指的是被继承的类，称为“基类”、“父类”或“超类”

继承真实地反映了客观世界中事物的层次关系，通过类的继承，能够实现对问题的深入抽象描述，反映出事物的发展过程。继承性是面向对象程序设计语言不同于其他语言的最主要的特点。

2.1.6多态

多态性是指类中同一函数名对应多个功能相似的不同函数，可以使用相同的调用方式来调用这些具有不同功能的同名函数，这些同名的函数可以是参数的个数或是类型不同，但是函数名相同，当进行调用的时候，根据所传的数据选定相应的函数，从而去执行不同的功能。

2.1.7消息

对象之间是通过消息进行通信的，多个对象之间通过传递消息来请求或提供服务，从而使一个软件具有更强大的功能。

在面向对象的系统中，把“请求”或“命令”抽象成“消息”，当系统中的其他对象请求这个 对象执行某个服务时，就将一个消息发送给另一个对象，接收到消息的对象将消息进行解释，然后响应这个请求，完成指定的服务。通常，把发送消息的对象称为发送者，把接收消息的对象称为接收者。通常，一个消息由以下几部分组成。

（1）提供服务的对象名。

（2）服务的标识，即方法名。

（3）输入信息，即实际参数。

（4）响应结果，即返回值或操作结果。

消息是实现对象之间进行通信的一种机制，对于一个对象可以接收不同形式的多个消息，并产生不同的结果；相同形式的消息可以发送给不同的对象，并产生不同的结果；在发送消息的时候可以不考虑具体的接收者，对象可以对消息做出响应，也可以拒绝消息，也就是说不是必须要对消息做出响应。

问题：面向对象技术中，封装性是一种（）

1. 封装技术
2. 信息隐蔽技术
3. 组合技术
4. 传递技术

答案：A

封装就是把一个对象的方法和属性组合成一个独立的单位，并尽可能隐蔽对象的属性、方法和实现细节的过程，仅仅将接口进行对外公开。

2.2面向对象开发

如果遵照面向对象方法的思想进行软件系统的开发，其过程共分成以下4个阶段。

（1）系统调査和需求分析，分析问题并求解。

（2）整理问题：对第一阶段的结果进一步抽象、归类整理。

（3）程序实现。

（4）系统测试。

2.1.1系统调查和需求分析

面向对象的系统调查和需求分析阶段主要是提取系统的需求，也就是要分析出为了满足用户的需求，系统必须“做什么”（系统能提供的功能），而不是“怎么做”（系统如何实现）。

2.2.2面向对象分析方法

面向对象的分析方法，指的是按照面向对象的概念和方法，在对任务的分析中，根据客观存在的事物以及事物之间的关系，归纳出相关的对象，包括对象的属性、行为及对象之间的联系，并将具有共同属性和行为的对象用一个类来表示。

2.2.3 面向对象设计方法

面向对象的设计方法是面向对象方法中的一个中间过渡环节。其主要作用是对OOA分析的结果进行规范化的整理，以便为面向对象程序设计阶段打下基础

问题：面向对象方法的开发过程是（）

1. 先从面向对象设计（OOD）到面向对象分析（OOA）再到面向对象编程（OOP）
2. 先从面向对象分析（OOA）到面向对象设计（OOD）再到面向对象编程（OOP）
3. 先从面向对象编程（OOP）到面向对象设计（OOD）再到面向对象分析（OOA）
4. 先从面向对象编程（OOP）到面向对象分析（OOA）再到面向对象设计（OOD）

　　答案：B

2.3软件建模概述

每个系统都可以从不同的方面用不同的模型来描述，因而每个模型都是一个在语义上闭合的系统抽象。模型可以是结构性的，强调系统的组织。它也可以是行为性的，强调系统的动态方面。

建模是为了能够更好地理解正在开发的系统。通过建模，要达到如下4个目的。

1. 模型有助于按照实际情况或按照所需要的样式对系统进行可视化。
2. 模型能够规约系统的结构或行为。
3. 模型给出了指导构造系统的模板。
4. 模型对做出的决策进行文档化。

2.3.1软件建模的概念

为建立复杂的软件系统，必须抽象出系统的不同视图，使用精确的符号建立模型，验证这些模型是否满足系统的需求，并逐渐添加细节信息把这些模型转变为实现。这就是软件建模。这样的一个过程就是软件模型形成的过程，软件建模是捕捉系统本质的过程，把问题领域转移到解 决领域的过程。

软件建模是开发优秀软件的一个核心工作，其目的是把要设计的结构和系统的行为联系起来，并对系统的体系结构进行可视化和控制。

2.3.2软件建模的用途

软件模型就是对复杂问题进行分层，从而更好地解决问题。有效的软件模型有利于分工与专业化生产，从而节省生产成本。为了降低软件的复杂程度，便于提早看到软件的将来，便于设计人员和开发人员交流从而使用了软件建模技术。

2.3.3软件建模的优点

软件建模主要有以下几个优点。

1. 使用模型便于从整体上、宏观上把握问题，以便更好地解决问题。
2. 软件建模可以加强软件工作人员之间的沟通，便于提早发现问题。
3. 模型为代码生成提供依据，帮助人们按照实际情况对系统进行可视化。
4. 模型允许人们详细说明系统的结构或行为，给出了一个指导人们构造系统的模板, 并对人们做出的决策进行文档化。

问题：什么是软件建模？

1. 开发软件模型
2. 再编码之前设计软件应用
3. 开发软件图
4. 开发软件原型

答案：B

**第3章 UML建模工具简介**

软件项目把软件开发维护过程中的需求分析、 系统结构设计、代码实现、系统测试及系统改进各个环节都进行了规范化。而UML就是为此而设计的一种图形化描述工具。所以说，实现UML的工具就会充分体现结构简明，容易理解，标准清楚。

3.1常用UML建模工具

面向对象的软件建模工具应对软件系统的模型进行可视化、构造和文档化。一套面向对象的软件建模工具应该给予特定的概念和表示方法，通过对建模人员进行过程性支持、辅 助进行建模外，还要安装规范生产相应的开发文档，尽可能多地生成代码。面向对象的软件建模工具应该具有以下功能。

1. 绘图。
2. 存储。
3. 一致性检査。
4. 对模型进行组织。
5. 导航。
6. 写作支持。
7. 代码生成。
8. 逆向项目。
9. 集成。
10. 支持多种抽象层和开发过程。
11. 文档生成。
12. 脚本编程。

在UML的发展中有很多工具被使用，其中比较有代表性的有Rational Rose、 PowerDesigner等，这里提出4种工具加以介绍。

3.1.1Rational Rose

Rational Rose是Rational公司出品的一种面向对象的统一建模语言的可视化建模工具。用于可视化建模和公司级水平软件应用的组件构造。它包括了统一建模语言（UML），OOSE，以及OMT。其中统一建模语言（UML）由Rational公司3位世界级面向对象技术专家Grady Booch、Ivar Jacobson、和Jim Rumbaugh通过对早期面向对象研究和设计方法的进一步扩展而得来的，它为可视化建模软件奠定了坚实的理论基础。  
Rational Rose 是一个完全的、具有能满足所有建模环境（Web开发，数据建模，Visual Studio和 C++ ）灵活性需求的一套解决方案。Rose 允许开发人员，项目经理，系统工程师和分析人员在软件开发周期内在将需求和系统的体系架构转换成代码，消除浪费的消耗，对需求和系统的体系架构进行可视化，理解和精练。通过在软件开发周期内使用同一种建模工具可以确保更快更好的创建满足客户需求的可扩展的、灵活的并且可靠的应用系统。  
Rational Rose不是单纯的绘图工具，它专门支持UML的建模，有很强的校验功能，能检査出模型中的许多裸机错误，还支持多种语言的双向项目，特别是对当前比较流行的 Java的支持非常好。Rose早期没有对数据库端建模的支持，但现在的版本中已经加入数据 库建模的功能。

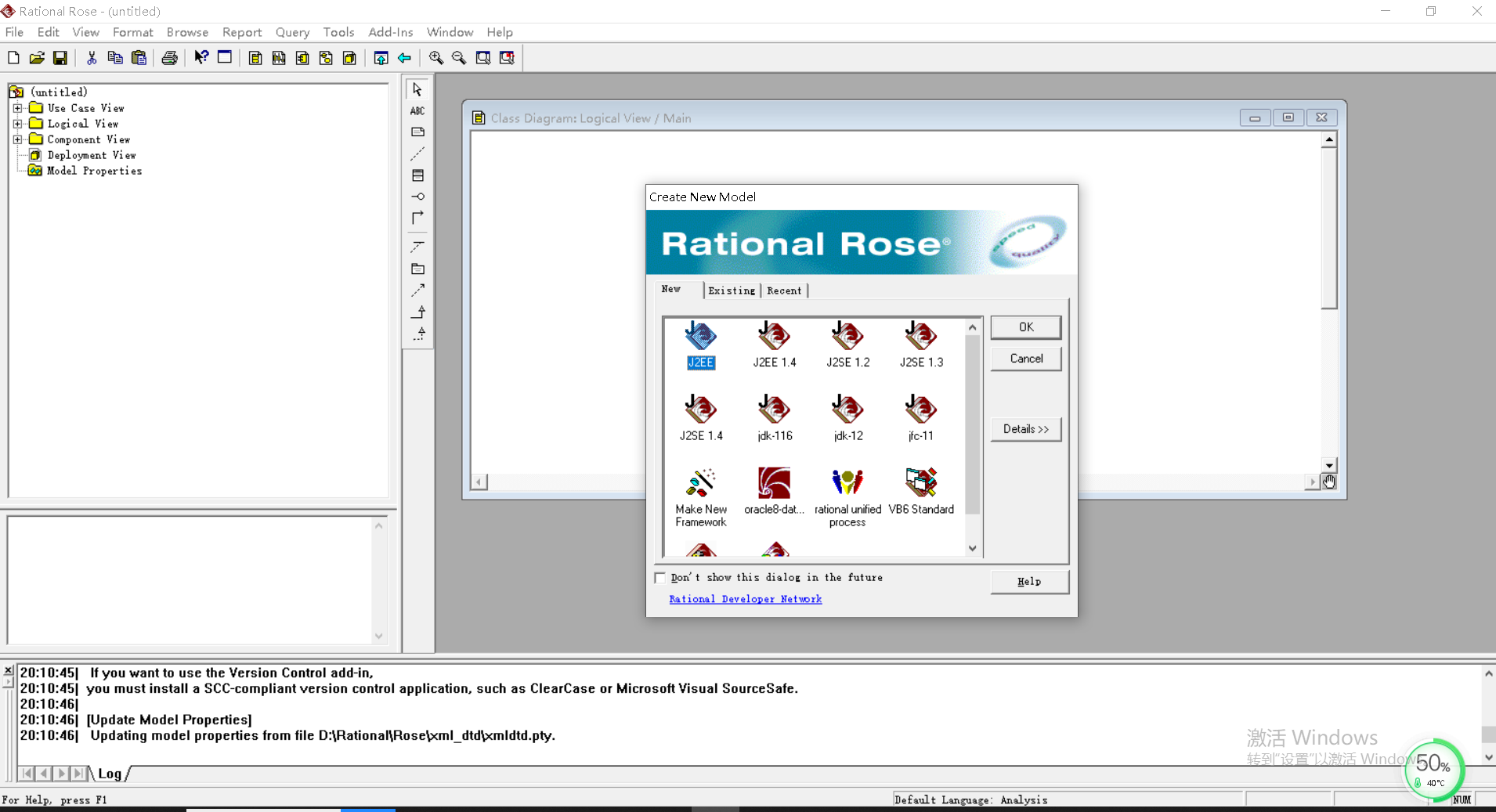
Rational Rose包含多个版本。

1. Rose Enterprise:支持用 C++、Java、Visual Basic和Oracle生成代码，支持逆向

项目。

1. Rose Professional系列：可以用一种语言生成代码。
2. Rose Modeler：可以对系统生成模型，但不支持逆向项目，也不支持由模型转出代码。

Rose Enterprise的设计窗口：

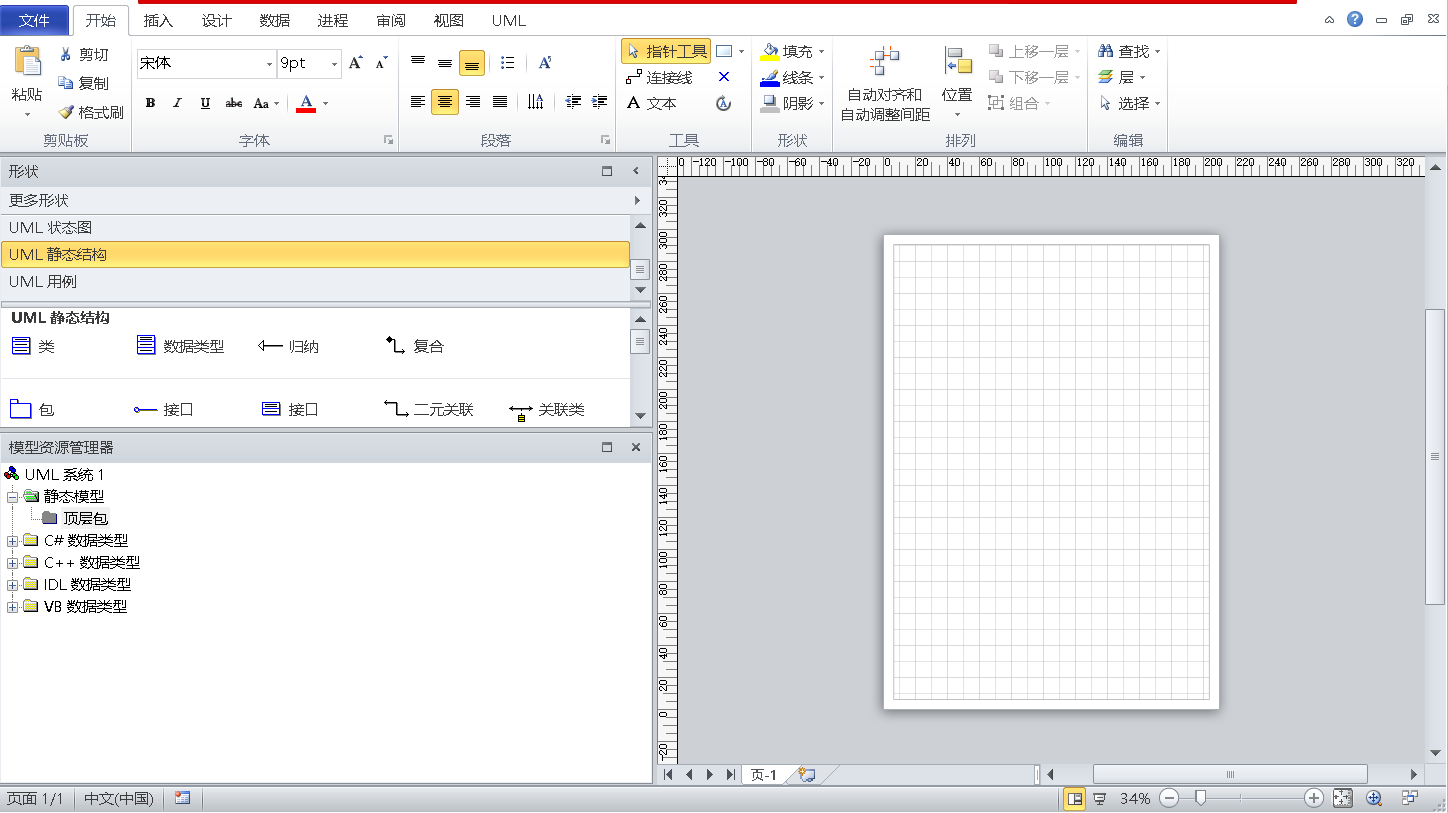


常用UML建模工具Rose具有强大的正向和逆向工程能力。正向工程这里指的是由设计产生代码，逆向工程指由代码归纳出设计。通过逆向工程Rose可以对历史系统作出分析，然后进行改进，再通过正向工程产生新系统的代码，这样的设计方式我们称之为再工程。

3.1.2Visio

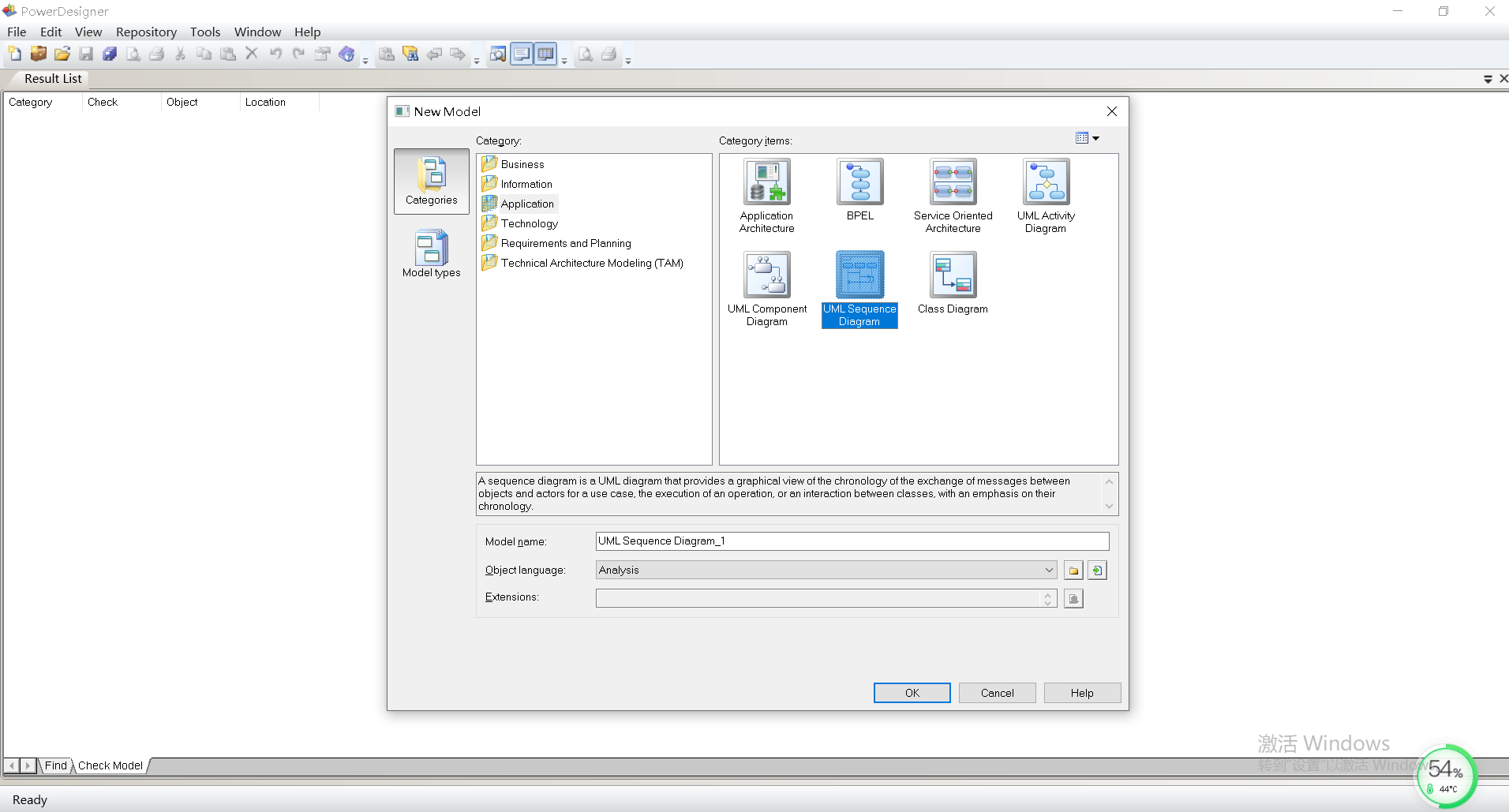
Office Visio 是office软件系列中的负责绘制流程图和示意图的软件，是一款便于IT和商务人员就复杂信息、系统和流程进行可视化处理、分析和交流的软件。使用具有专业外观的 Office Visio 图表，可以促进对系统和流程的了解，深入了解复杂信息并利用这些知识做出更好的业务决策，可以提供了各种模板。

VISIO原来仅仅是一种画图工具，能够用来描述各种图形（从电路图到房屋结构图），也是到VISIO2000才开始引进软件分析设计功能到代码生成的全部功能，可以说是目前最能够用图形方式来表达各种商业图形用途的工具（对软件开发中的UML支持仅仅是其中很少的一部分）。它跟微软的office产品的能够很好兼容。能够把图形直接复制或者内嵌到WORD的文档中。但是对于代码的生成更多是支持微软的产品如VB，VC++，C#，MS SQL Server 等（这也是微软的传统），所以它可以说用于图形语义的描述比较方便，但是用于软件开发过程的迭代开发则有点牵强。



3. 1. 3 PowerDesigner

Power Designer 是Sybase公司的CASE工具集，使用它可以方便地对管理信息系统进行分析设计，他几乎包括了数据库模型设计的全过程。利用Power Designer可以制作数据流程图、概念数据模型、物理数据模型，还可以为数据仓库制作结构模型，也能对团队设计模型进行控制。他可以与许多流行的软件开发工具，例如PowerBuilder、Delphi、VB等相配合使缩短开发时间和使系统设计更优化。  
 Power Designer是能进行数据库设计的强大的软件，是一款开发人员常用的数据库建模工具。使用它可以分别从概念数据模型(Conceptual Data Model)和物理数据模型(Physical Data Model)两个层次对数据库进行设计。在这里，概念数据模型描述的是独立于数据库管理系统(DBMS)的实体定义和实体关系定义；物理数据模型是在概念数据模型的基础上针对目标数据库管理系统的具体化。



问题：UML是下面哪种方法的建模工具：

1. 结构化方法
2. 原型化方法
3. CASE方法
4. 面向对象方法

答案：C

问题：Power Designer可以从哪两个层次对数据库进行设计？

答案：可以分别从概念数据模型(Conceptual Data Model)和物理数据模型(Physical Data Model)两个层次对数据库进行设计

3. 1. 4 StarUML

StarUML(简称SU),是一款开放源代码的UML开发工具，可以直接到StarUML网站下载。

StarUML是一种创建UML类图，生成类图和其他类型的统一建模语言(UML)图表的工具。StarUML发展快、灵活、可扩展性强。

1. 可绘制UML中的常用图

UML2. 0 分为两大类：结构图(Structure Diagram)和行为图(Behavior Diagram)共 13 种图。结构图用于对系统的静态结构建模，包括类图、组合结构图、构件图、部署图、对象图和包图；行为图用于对系统的动态行为建模，包括实例图、交互图(顺序图、通信图、交互概 览图、计时图)、活动图和状态机图。StarUML可支持这些图的绘制。

1. 完全免费

StarUML是一套开放源码的软件，不仅免费自由下载.连代码都免费开放。

1. 多种格式

StarUML遵守UML的语法规则，不支持违反语法的动作。

1. 双向工程

StarUML可以依据类图的内容生成Java、C++、C#代码，也能够读取Java、C++、C#代码反向生成类图。逆向工程有两个主要用途，其一，是就有的源码反转成图之后，可以构建 UML模型的方式继续将新的设计添加上去；其二，是想要解析源码时，可以通过反转的类图来理解，不再需要查看一行又一行的代码，这将节省大量的时间和精力。

1. 支持**XMI**

StarUM L 接受 XMI 1. 1,1. 2 和 1. 3 版的导入导出。XMI ( XML-based Metadata Interchange)是一种以XML为基础的交换格式，用以交换不同开发工具所生成的UML 模型。

1. 导入**Rose**文件

StarUML可以读取Rational Rose生成的文件，让原先Rose的用户可以转而使用免费 的StarUMLo

1. 支持模式

支持23种GoF模式(Pattern),以及三种EJB模式。

StarUML设计窗口：

