

UML 模型图

1. UML 的定义

作为一种建模语言，UML 的定义包括 UML 语义和 UML 表示法两个部分。

- UML 语义：描述基于 UML 的精确元模型定义。
- UML 表示法：定义 UML 符号的表示法，为开发者或开发工具使用这些图形符号和文本语法为系统建模提供了标准。这些图形符号和文字所表达的是应用级的模型，在语义上它是 UML 元模型的实例。

2. 五类图定义。

2.1. 用例图：

从用户角度描述系统功能，并指出各功能的操作者。

2.2. 静态图：包括类图和对象图

类图描述系统中类的静态结构，不仅定义系统中的类，表示类之间的联系，如关联、依赖、聚合等，也包括类的属性和操作，类图描述的是一种静态关系，在系统的整个生命周期都是有效的。对象图是类图的实例，几乎使用与类图完全相同的标识。一个对象图是类图的一个实例。由于对象存在生命周期，因此对象图只能在系统某一段时间段存在。

2.3. 行为图：

描述系统的动态模型和组成对象间的交互关系，包括状态图和活动图。状态图描述类的对象所有可能的状态以及事件发生时状态的转移条件，状态图是对类

图的补充，活动图描述满足用例要求所要进行的活动以及活动间的约束关系，有利于识别并进行活动。

2.4. 交互图：

描述对象间的交互关系，包括**时序图**和**协作图**。时序图显示对象之间的动态合作关系，它强调对象之间消息发送的顺序，同时显示对象之间的交互；协作图描述对象间的协作关系，协作图跟时序图相似，显示对象间的动态合作关系。除显示信息交换外，协作图还显示对象以及它们之间的关系。如果强调时间和顺序，则使用时序图；如果强调上下级关系，则选择协作图。

2.5. 实现图：包括组件图和部署图。

组件图描述代码部件的物理结构及各部件之间的依赖关系，组件图有助于分析和理解部件之间的相互影响程度；部署图定义系统中软硬件的物理体系结构。

采用 **UML** 来设计系统时，第一步是描述需求；第二步根据需求建立系统的静态模型，以构造系统的结构；第三步是描述系统的行为。其中在第一步与第二步中所建立的模型都是静态的，包括用例图、类图、对象图、组件图和部署图等 **5** 种图形，是标准建模语言 **UML** 的静态建模机制。其中第三步中所建立的模型或者可以执行，或者表示执行时的时序状态或交互关系。它包括状态图、活动图、时序图和协作图等 **4** 种图形，是标准建模语言 **UML** 的动态建模机制。

3. UML 九种图的作用简介

3.1. 用例图

描述角色以及角色与用例之间的连接关系。说明的是谁要使用系统，以及他们使用该系统可以做些什么。一个用例图包含了多个模型元素，如系统、参与者和用例，并且显示了这些元素之间的各种关系，如泛化、关联和依赖。

3.2. 类图

类图是描述系统中的类，以及各个类之间的关系的静态视图。能够让我们在正确编写代码以前对系统有一个全面的认识。类图是一种模型类型，确切的说，是一种静态模型类型。

3.3. 对象图

与类图极为相似，它是类图的实例，对象图显示类的多个对象实例，而不是实际的类。它描述的不是类之间的关系，而是对象之间的关系。

3.4. 活动图

描述用例要求所要进行的活动，以及活动间的约束关系，有利于识别并行活动。能够演示出系统中哪些地方存在功能，以及这些功能和系统中其他组件的功能如何共同满足前面使用用例图建模的商务需求。

3.5. 状态图

描述类的对象所有可能的状态，以及事件发生时状态的转移条件。可以捕获对象、子系统和系统的生命周期。他们可以告知一个对象可以拥有的状态，并且事件(如消息的接收、时间的流逝、错误、条件变为真等)会怎么随着时间的

推移来影响这些状态。一个状态图应该连接到所有具有清晰的可标识状态和复杂行为的类；该图可以确定类的行为，以及该行为如何根据当前的状态变化，也可以展示哪些事件将会改变类的对象的状态。状态图是对类图的补充。

3.6. 序列图（顺序图）

序列图是用来显示你的参与者如何以一系列顺序的步骤与系统的对象交互的模型。顺序图可以用来展示对象之间是如何进行交互的。顺序图将显示的重点放在消息序列上，即强调消息是如何在对象之间被发送和接收的。

3.7. 协作图

和序列图相似，显示对象间的动态合作关系。可以看成是类图和顺序图的交集，协作图建模对象或者角色，以及它们彼此之间是如何通信的。如果强调时间和顺序，则使用序列图；如果强调上下级关系，则选择协作图；这两种图合称为交互图。

3.8. 构件图（组件图）

描述代码构件的物理结构以及各种构建之间的依赖关系。用来建模软件的组件及其相互之间的关系，这些图由构件标记符和构件之间的关系构成。在组件图中，构件是软件单个组成部分，它可以是一个文件，产品、可执行文件和脚本等。

3.9. 部署图（配置图）

是用来建模系统的物理部署。例如计算机和设备，以及它们之间是如何连接的。部署图的使用者是开发人员、系统集成人员和测试人员。

4. 几种图的区别

4.1. 这九种模型图各有侧重，

- 1:用例图侧重描述用户需求，
- 2:类图侧重描述系统具体实现；

4.2. 描述的方面都不相同，

- 1:类图描述的是系统的结构，
- 2:序列图描述的是系统的行为；

4.3. 抽象的层次也不同，

- 1： 构件图描述系统的模块结构，抽象层次较高，
- 2： 类图是描述具体模块的结构，抽象层次一般，
- 3： 对象图描述了具体的模块实现，抽象层次较低。

4.4. 补充说明：

在有的文献书籍中，将这九种模型图分为三大类：

结构分类、动态行为和模型管理：

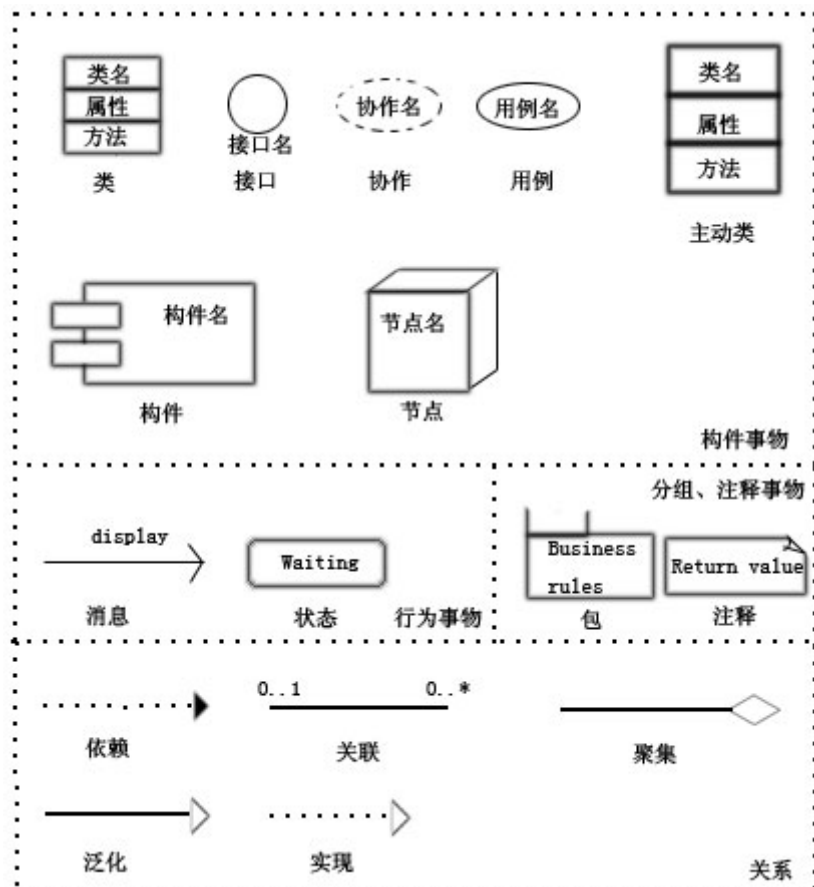
- 1： 结构分类包括用例图、类图、对象图、构件图和部署图，
- 2： 动态行为包括状态图、活动图、顺序图和协作图，
- 3： 模型管理则包含类图。

5. 画图详述 UML 图例

5.1. 总体分析

UML(统一建模语言): 是面向对象的可视化建模的一种语言。是数据库设计过程中, 在 E-R 图 (实体-联系图) 的设计后的进一步建模。

UML 中有 3 种构造块: 事物、关系和图, 事物是对模型中最具有代表性的成分的抽象; 关系是把事物结合在一起; 图聚集了相关的事物。具体关系图标如下:



说明:

构件事物是名词，是模型的静态部分。

行为事物是动态部分，表示行为。

分组事物是组织部分。

注释事物是解释部分。

依赖：一个事物变化会引起另一个事物变化。

聚集：特殊的关联，描述整体与部分的组合关系。

泛化：是一种特殊与一般的关系，如子元素（特殊）与父元素（一般），箭头指向父元素。

实现：类元之间的关系，其中一个类元指定了由另一个类元保证执行的契约。一般用在接口和实现他们的类之间或用例和实现它们的协作之间。

UML 提供 9 种视图：类图、对象图，用例图，序列图、协作图，状态图、活动图，构件图和部署图。

在 UML 系统开发中有三个主要的模型：

功能模型: 从用户的角度展示系统的功能，包括用例图。

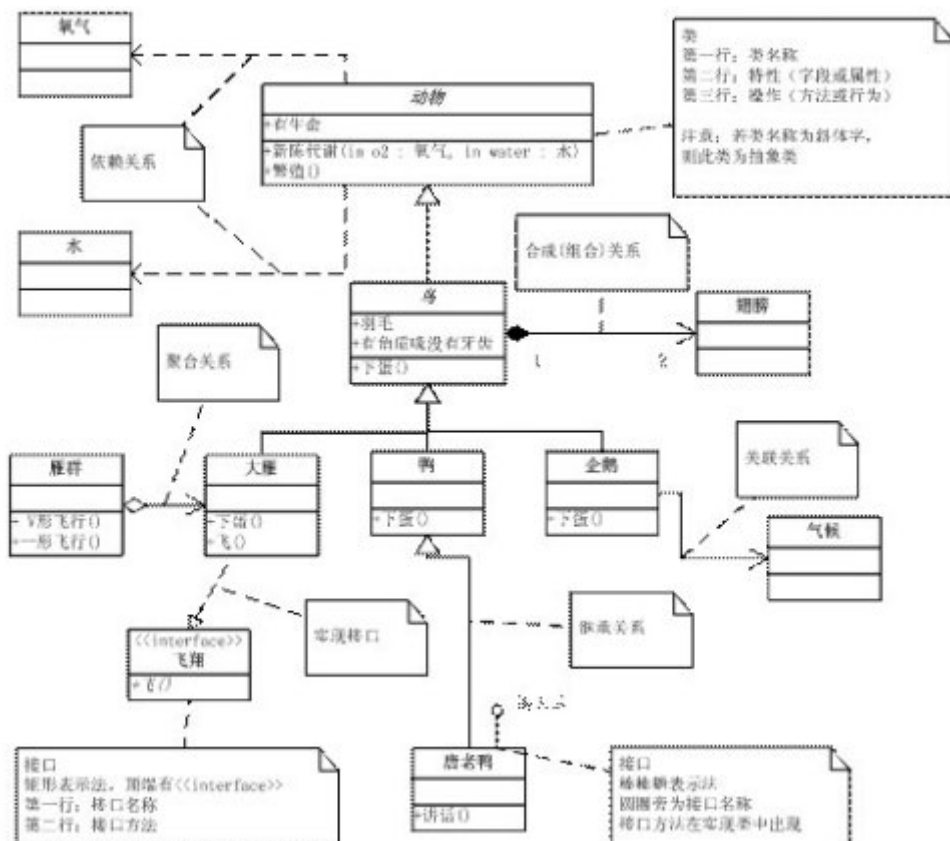
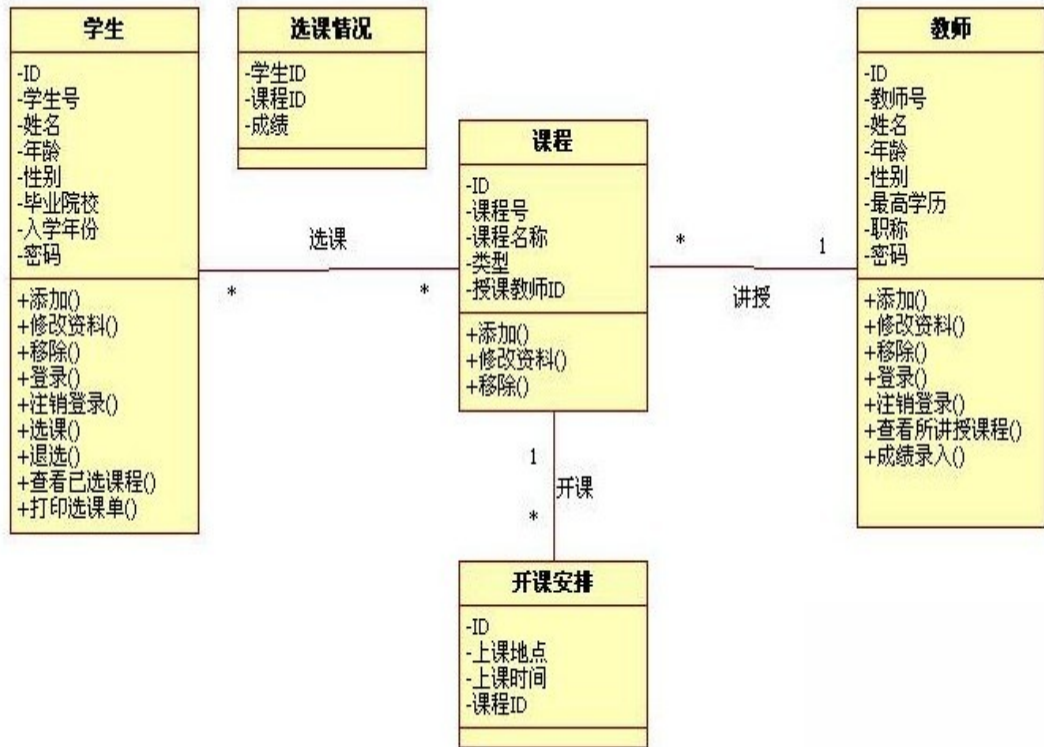
对象模型: 采用对象，属性，操作，关联等概念展示系统的结构和基础，包括类图。

动态模型: 展现系统的内部行为。包括序列图，活动图，状态图。

5.2. 图解九种 UML 图

1). 类图:

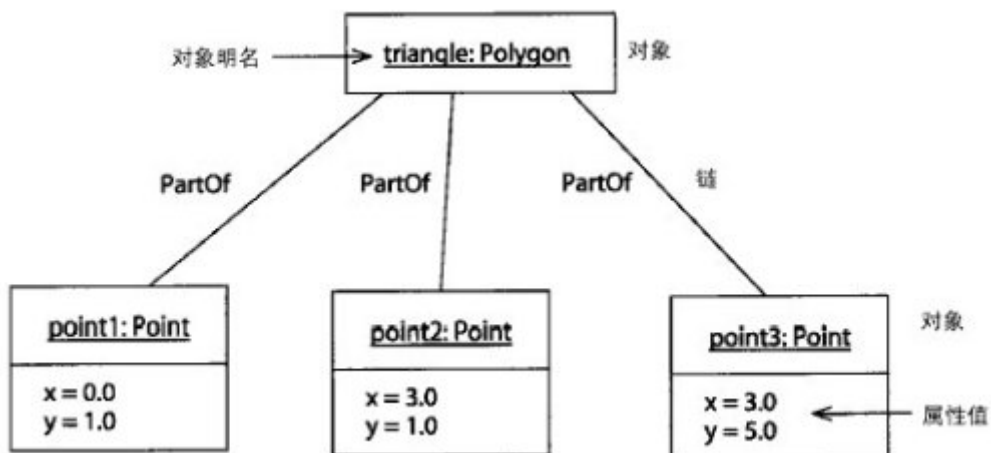
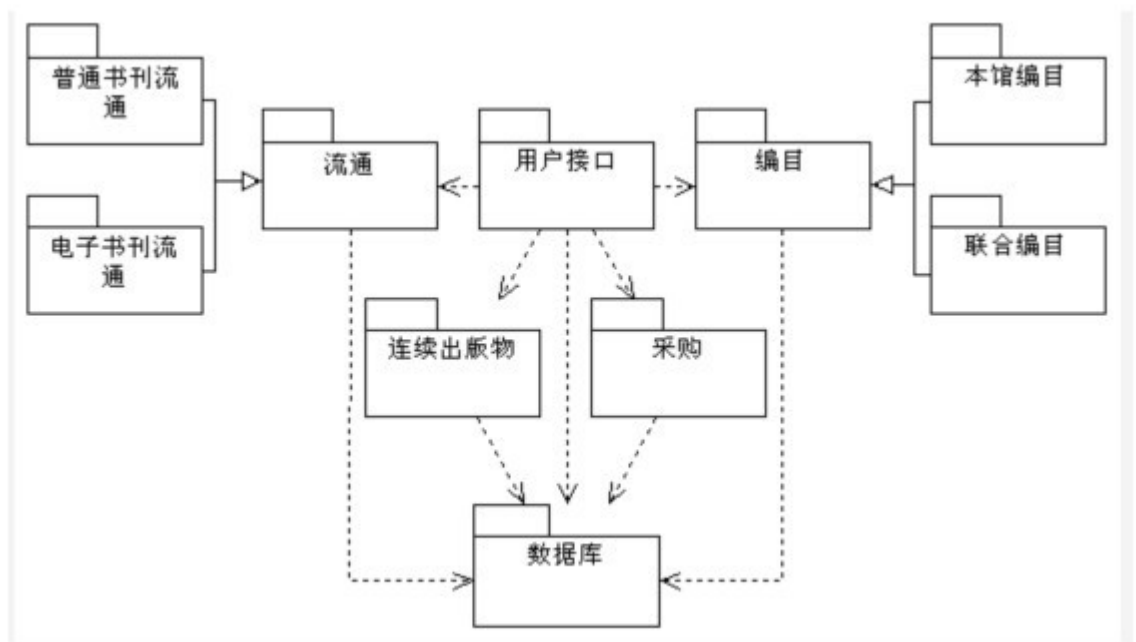
描述一组对象、接口、协作等事物之间的关系



注：#表示 protected，+表示 Public，-表示 private

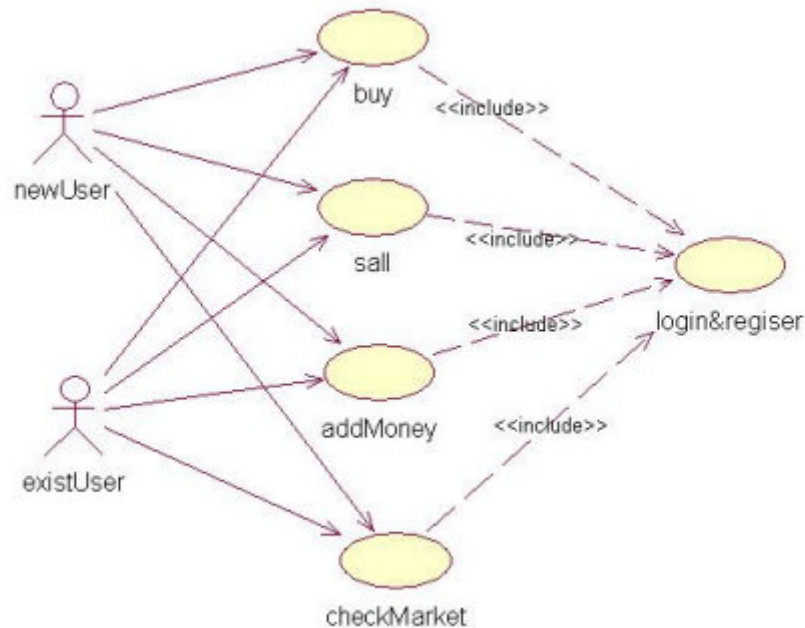
2). 对象图:

描述一组对象之间的关系，是具有具体属性值和行为的一个具体事物，其是类图中所建事物实例的静态快照，其与类图的主要区别是一个是抽象的,而对象图是具体的。



3).用例图:

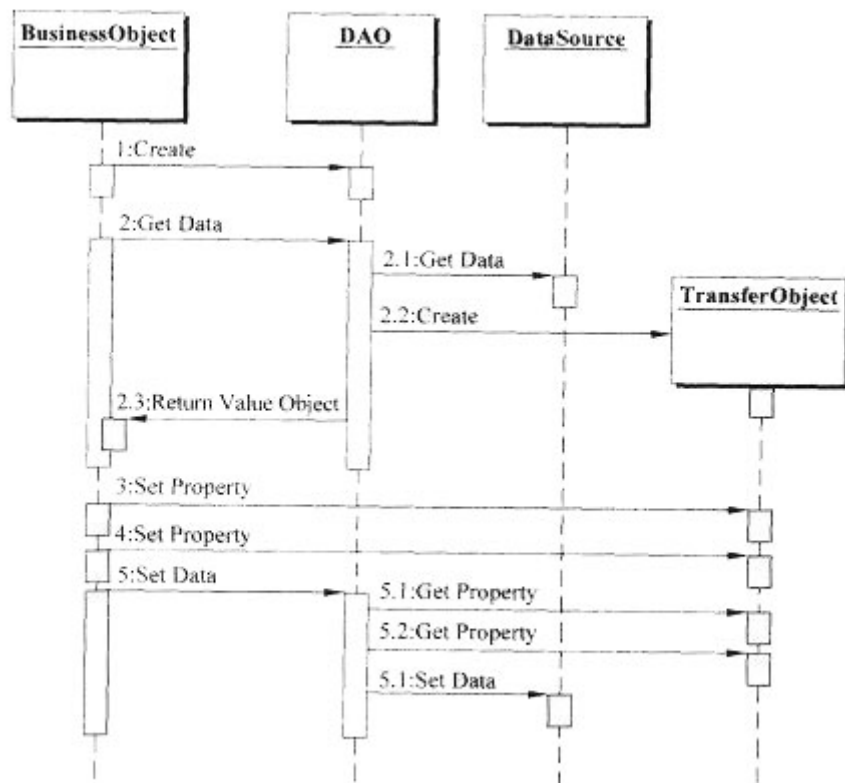
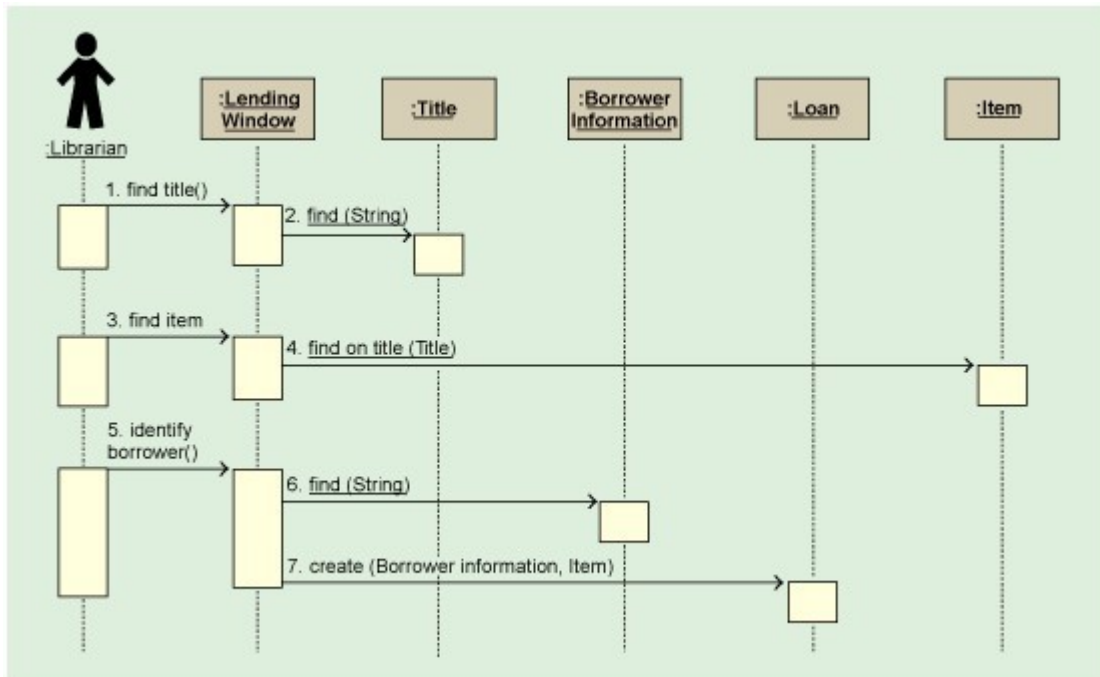
描述一组用例、参与者以及它们之间的关系，其展示的是该系统在它的外面环境中所提供的外部可见服务。如下图



4).交互图:

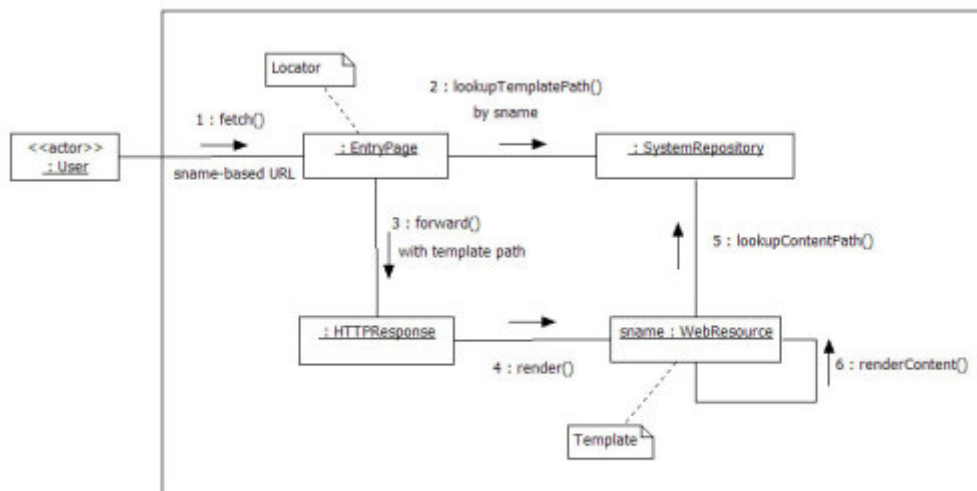
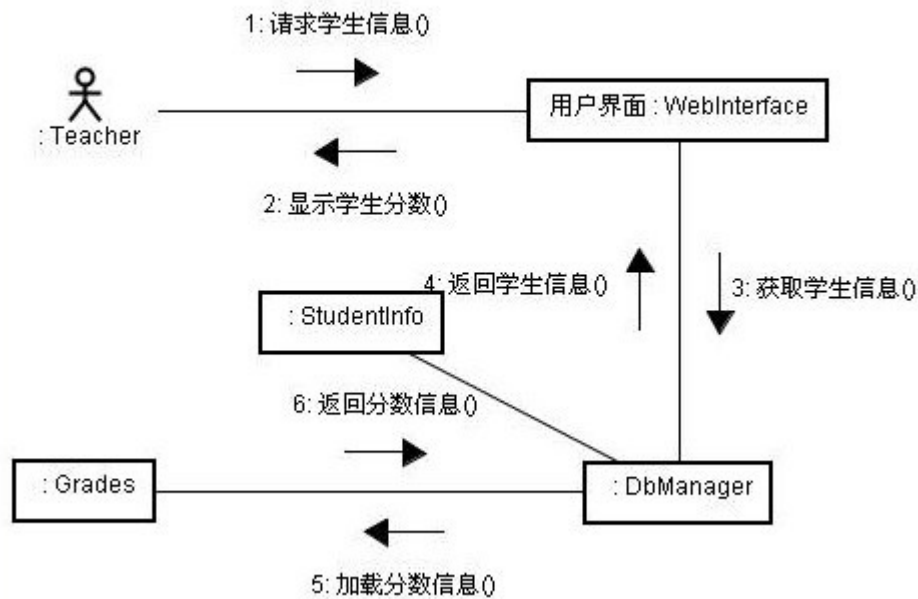
包括序列图（顺序图）和协作图，两者对应，顺序图是强调消息时间顺序，有对象生命线和控制焦点。协作图是强调接收和发送消息的对象的结构组织，有路径和顺序号。如下图

4.1). 序列图:



4. 2). 协作图:

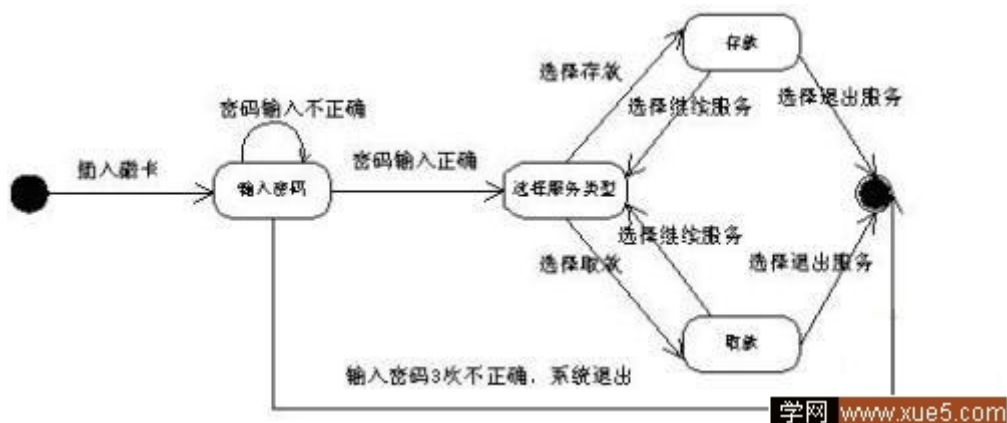
例：



SimpleNameLocator 实现协作图（设计模型）
(c) 2007 zhangxun.com

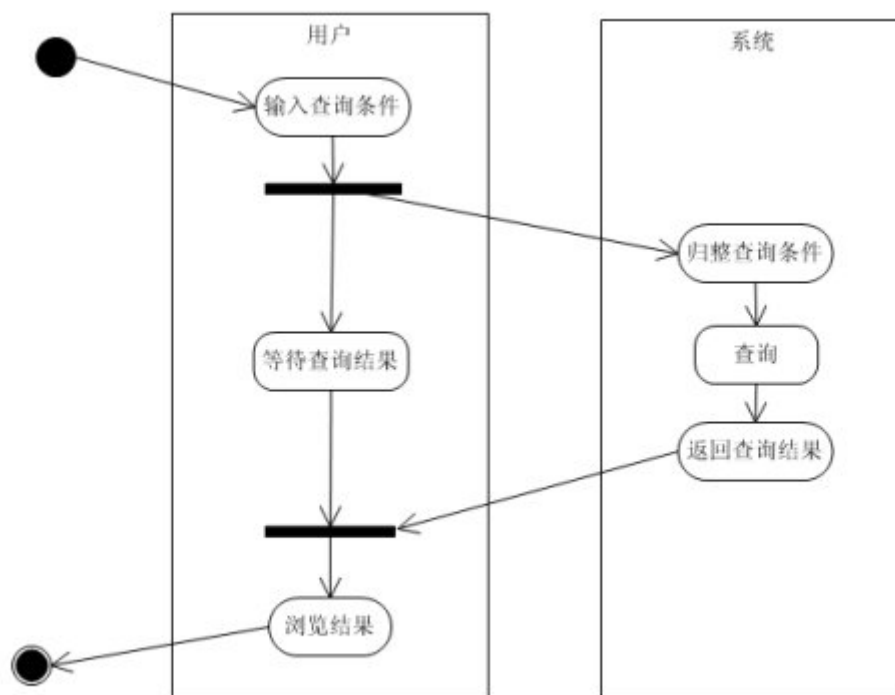
5). 状态图：

展示了一个状态机，由状态、转换、事件和活动组成。强调事件行为的顺序。如下图



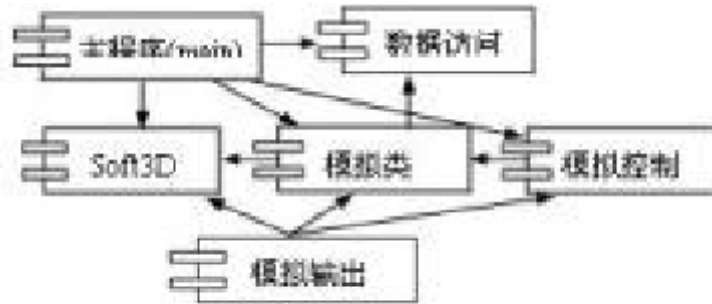
6). 活动图：

是一种特殊的状态图，实现一个活动到另一个活动的流程。如下图



7). 构件图

构件图展示一组构件之间的组织和依赖关系，并以全局的模型展示出来。



8). 部署图:

部署图是构件的配置及描述系统如何在硬件上部署。如下图

