



交互图

内容

- 交互图概述
- 时序图
- 时序图中的消息

交互图概述

- 交互图(interaction diagram)的概念
 - 用来描述对象之间以及对象与参与者之间的动态协作关系以及协作过程中行为次序的图形文档.
- 交互图的作用
 - 通常用于描述一个用例的行为, 显示该用例中所涉及的对象及对象之间消息传递情况.
- 交互图的类型
 - 时序图
 - 协作图

时序图的基本概念

1. 时序图的定义

- 在UML的表示中, 时序图将交互关系表示为一个二维图。其中, 纵向是时间轴, 时间沿竖线向下延伸。横向代表了在协作中各独立对象的角色。角色使用生命线进行表示, 当对象存在时, 生命线用一条虚线表示, 此时对象不处于激活状态, 当对象的过程处于激活状态时, 生命线是一个双道线。时序图中的消息使用从一个对象的生命线到另一个对象生命线的箭头表示, 箭头以时间顺序在图中从上到下排列。

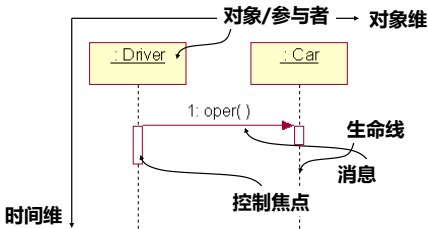
时序图的基本概念

2. 时序图在项目开发里的作用

- 时序图作为一种描述在给定语境中消息是如何在对象间传递的图形化方式, 在使用其进行建模时, 主要可以将其用途分为以下三个方面:
 - (1)确认和丰富一个使用语境的逻辑表达。一个系统的使用情境就是系统潜在的使用方式的描述, 也就是它的名称所要描述的。一个使用情境的逻辑可能是一个用例的一部分, 或是一条控制流。
 - (2)细化用例的表达。我们前面已经提到, 时序图的主要用途之一, 就是把用例表达的需求, 转化为进一步、更加正式层次的精细表达。用例常常被细化为一个或者更多的时序图。
 - (3)有效地描述如何分配各个类的职责以及各类具有相应职责的原因。我们可以根据对象之间的交互关系来定义类的职责, 各个类之间的交互关系构成一个特定的用例。例如, “Customer对象向Address对象请求其街道名称”指出Customer对象应该具有“知道其街道名”这个职责。

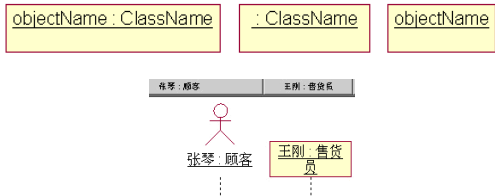
时序图

时序图的样式和元素



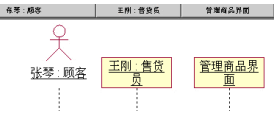
时序图

- 时序图的样式和元素
 - 对象的三种命名方式



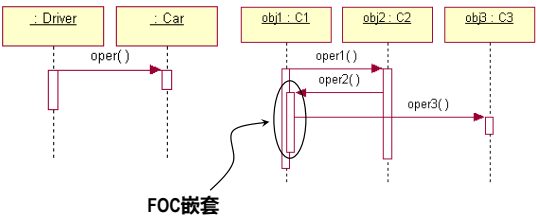
时序图

- 时序图的样式和元素
- 生命线：一条垂直的虚线，用来表示时序图中的对象在一段时间内的存在。



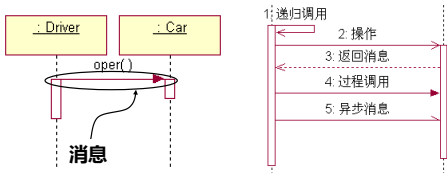
时序图

- 时序图的样式和元素
 - 控制焦点 (Focus of Control): 又称为激活期 (Activation), 用生命线上的小矩形表示, 在这个时间段内, 对象执行相应的操作。



时序图

- 时序图的样式和元素
 - 消息：对象之间传输的信息

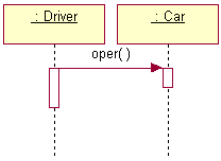


时序图中的消息

- 调用消息 `procedure call`
- 异步消息 `asynchronous`
- 返回消息 `return`
- 阻止消息 `balking`
- 超时消息 `time-out`

时序图中的消息

调用消息：消息的发送者把控制传递给消息的接收者，等待接收者返回或放弃控制。



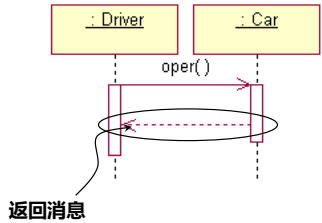
调用消息必有一个与之配对的返回消息，但是可以不用画出。

时序图中的消息

异步消息： 消息的发送者把控制传递给消息的接收者，然后继续自己的工作，不等待接收者返回或放弃控制。

时序图中的消息

返回消息： 表示从过程调用返回。如果是过程消息返回，则不用画出；如果是非过程消息，则必须画出。



时序图中的消息

阻止消息： 消息的发送者传递消息给接收者，如果接收者无法立即接收，则发送者放弃该消息。

超时消息： 消息的发送者发出消息给接收者并按指定时间等待，若接收者无法在指定时间内接收，则发送者放弃该消息。

时序图中的消息

消息的语法格式

[前置消息列表][警戒条件][消息序号][返回值:=]消息名称([参数列表])

例子:

2:display(x,y)	简单消息
1.3.1 p:= find(specs)	嵌套消息, 有返回值
[x<0] 4: invert(x, color)	条件消息
3.1 *: update()	循环消息
A3, B4/ C2: copy(a, b)	线程间同步

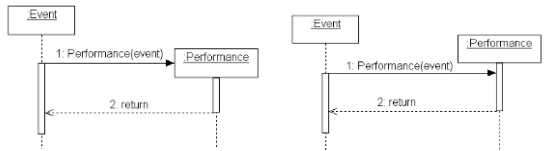
时序图

时序图的样式和元素

- 对象的创建和撤消
 - 如果对象位于时序图的顶部，说明在交互开始之前该对象已经存在了。如果对象是在交互的过程中创建的，那么它应当位于图的中间部分。
 - 对象在创建消息发生之后才能存在，对象的生命线也是在创建消息之后才存在的。

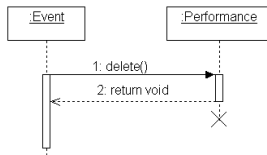
时序图

创建对象的两种表示方法:



时序图

如果要撤销一个对象，只要在其生命线终止点放置一个“X”符号即可，该点通常是对删除或取消消息的回应。

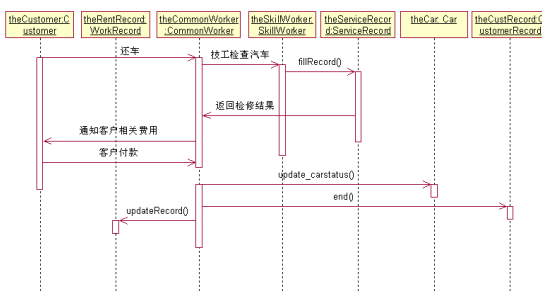


分支与从属流

在UML中，存在两种方式可以来修改时序图中消息的控制流，分别是：分支和从属流。

- 分支指的是从同一点发出多个消息并指向不同的对象，根据条件是否互斥，可以有条件和并行两种结构。由于时序图只表示某一个活动按照时间顺序的经历过程，所以在Rational Rose 2003中，对时序图的分支画法没有明显的支持。
- 从属流指的是从同一点发出多个消息指向同一个对象的不同生命线。从属流在Rational Rose 2003中不支持，因为添加从属流以后会明显增加时序图的复杂度。

汽车租赁系统中“客户还车”的时序图



建立时序图的步骤

1. 从用例中识别交互过程
2. 识别参与交互过程的对象
3. 为每个对象设置生命线，识别交互过程中对象的创建与删除
4. 从引发这个交互过程的初始消息开始，在生命线之间自顶向下依次画出随后的各个消息。
5. 如果需要说明时间约束，则在消息旁边加上约束说明。
6. 如果需要，可以为每个消息设置前置条件和后置条件。

使用Rose创建时序图

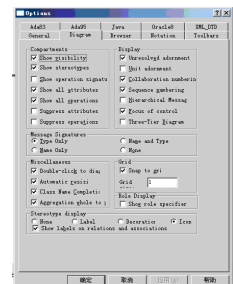
1. 创建对象

- 创建一个新的时序图，可以通过以下方式进行。
 - (1) 右键单击浏览器中的Use Case View (用例视图)、Logical View (逻辑视图) 或者位于这两种视图下的包。
 - (2) 在弹出的菜单中，选中“New” (新建) 下的“Sequence Diagram” (时序图) 选项。
 - (3) 输入新的序列名称。
 - (4) 双击打开浏览器中的时序图。

使用Rose创建时序图

2. 创建生命线

- 当对象被创建后，生命线便存在。当对象被激活后，生命线的一部分虚线变成细长的矩形框。
- 在Rational Rose 2003中，是否将虚线变成矩形框是可选的。我们可以通过菜单栏设置是否显示对象生命线被激活时的矩形框。
- 设置是否显示对象生命线被激活的矩形框步骤为：在菜单栏中选择“Tools” (工具) 下的“Options” (选项) 选项，在弹出的对话框中选择“Diagram” (图) 选项卡，选择或取消“Focus of control”选项。



使用Rose创建时序图

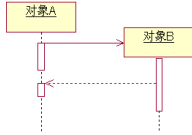
3. 创建消息

- 在时序图中添加对象与对象之间的简单消息的步骤如下：
 - 选择时序图的图形编辑工具栏中的图标，或者选择菜单栏“Tools”（工具）中“Create”（新建）下的“Object Message”选项，此时的光标变为“↑”符号。
 - 单击需要发送消息的对象。
 - 将消息的线段拖动到接收消息的对象中。
 - 在线段中输入消息的文本内容。
 - 双击消息的线段，弹出设置消息规范的对话框，在“General”选项卡中可以设置消息的名称，消息的名称也可以是消息接收对象的一个执行操作，我们在名称的下列菜单中选择一个或从新创建一个即可，称之为消息的绑定操作。
 - 如果需要设置消息的同步信息，也就是说设置消息成为简单消息、同步消息、异步消息、返回消息、过程调用、阻止消息和超时消息等，可以在“Detail”选项卡中进行设置，还可以设置以消息的频率。

使用Rose创建时序图

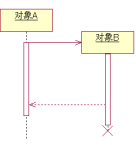
4. 创建对象与销毁对象

- 由于创建对象操作也是消息的一种，我们仍然可以通过发送消息的方式创建对象。在时序图的图形表示中，和其它对象不一样的，其它对象通常位于图的顶部，被创建的对象通常位于图的中间部位。

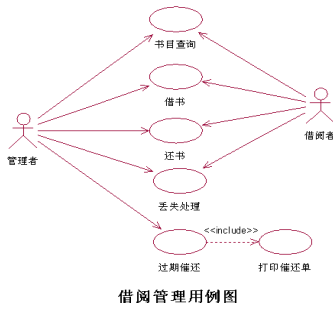


使用Rose创建时序图

- 销毁对象表示对象生命线的结束，在对象生命线中使用一个“X”来进行标识。给对象生命线上添加销毁标记的步骤如下：
 - 在时序图的图形编辑工具栏中选择 按钮，此时的光标变为“+”符号。
 - 单击欲销毁对象的生命线，此时该标记在对象生命线上标识。该对象生命线自销毁标记以下的部分消失。



实例：图书馆借书处理的时序图



实例：图书馆借书处理的时序图

- 用例：借书**
- 参与者：管理员, 借阅者**
- 操作流：**
 - 管理员进入图书馆借阅界面，用例开始。
 - 系统要求输入借阅者的借书证编码。
 - 系统检验借书证编码,如果正确,则显示借阅者的信息。
 - A1: 借书证编码有错。
 - A2: 如果该借阅者所借图书已经超期,则提示,本次拒借。
 - 系统要求输入所借图书的条码。
 - 系统显示所借图书的信息。
 - 确认借书。
 - 系统回到上一界面，等待处理下一借书。

实例：图书馆借书处理的时序图

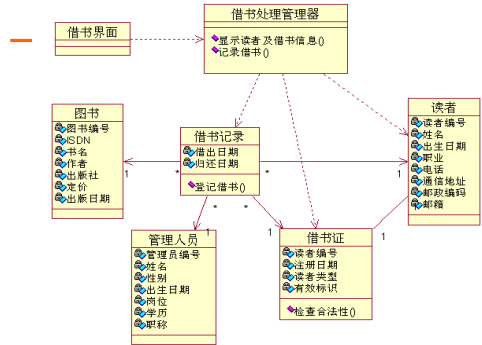
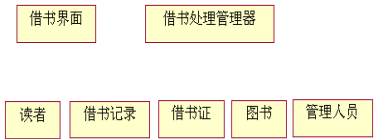
- 识别交互过程**

读者在借书时，先由管理员把借书证上的读者编号扫描给系统，系统检查这个读者的借书证信息是否合法。如果不合法,则系统给出提示,如果正确系统返回读者的身份信息，以及读者的借阅信息。

如果读者借书数量没有超过借书的上限，则把要借书的图书编号输入系统，系统登记借书信息，并返回借书成功信息，借书过程完成。

实例：图书馆借书处理的时序图

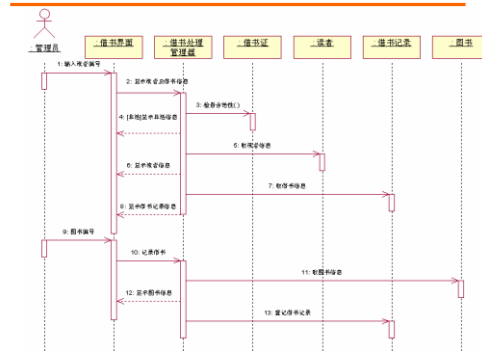
- 识别参与交互过程的对象



实例：图书馆借书处理的时序图

- 为每一个对象设置生命线,并确定对象的存在期限;
- 从引发交互的初始消息开始,在对象生命线上依次画出交互的消息
- 绘制时序图

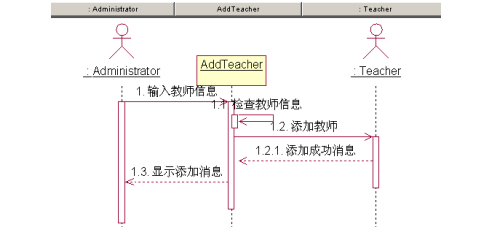
实例：图书馆借书处理的时序图



练习题

(1) 以“远程网络教学系统”为例,在该系统中,系统管理员需要登录系统才能进行系统维护工作,如添加教师信息、删除教师信息等。根据系统管理员添加教师信息用例,创建相关时序图。

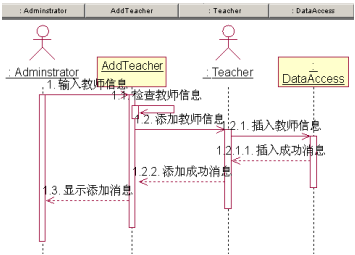
练习题



练习题

(2) 在“远程网络教学系统”中，如果我们单独抽象出来一个数据访问类来进行数据访问。那么，根据系统管理员添加教师信息用例，重新创建相关时序图。

练习题



填空题

- 时序图是由对象、生命线、控制焦点和_____等构成的。
- 消息

填空题

- 在UML的表示中时序图将交互关系表示为一张二维图，其中纵向是_____，时间沿竖线向下延伸。
- 时间轴

填空题

- UML的时序图中,横向代表了在协作中_____。
- 各独立对象的角色

填空题

- 时序图中，消息用_____表示。
- 水平箭头线

选择题

- 生命线是UML视图中哪个图形的组成部分（ ）
A. 类图
B. 状态图
C. 活动图
D. 时序图

● 选D

43

选择题

- 在UML的时序图中，通常由左向右分层排列各个对象，正确的排列方法是（ ）
A. 执行者角色,控制类,用户接口,业务层,后台数据库
B. 执行者角色,用户接口,控制类,业务层,后台数据库
C. 执行者角色,控制类,用户接口,后台数据库,业务层
D. 执行者角色,用户接口,业务层,控制类,后台数据库

● 选A

44

选择题

- 时序图由类角色，生命线，激活期和（ ）组成
A. 关系
B. 消息
C. 用例
D. 实体

● 选B

45

应用题

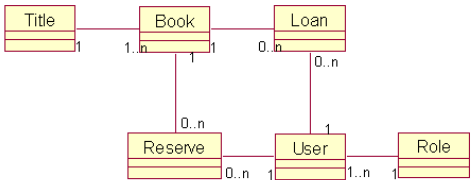
- 图书管理系统功能性需求说明如下：
图书管理系统能够为一定数量的借阅者提供服务。每个借阅者能够拥有唯一标识其存在的编号。图书馆向每一个借阅者发放图书证，其中包含每一个借阅者的编号和个人信息。提供的服务包括：提供查询图书信息、查询个人信息服务和预定图书服务等。
当借阅者需要借阅图书、归还书籍时需要通过图书管理员进行，即借阅者不直接与系统交互，而是通过图书管理员充当借阅者的代理和系统交互。
系统管理员主要负责系统的管理维护工作，包括对图书、数目、借阅者的添加、删除和修改。并且能够查询借阅者、图书和图书管理员的信息。
可以通过图书的名称或图书的ISBN/ISSN号对图书进行查找。

画出类图，并指明各个关系的重数。

对上一题的需求，画出语境“借阅者预定图书”的时序图。

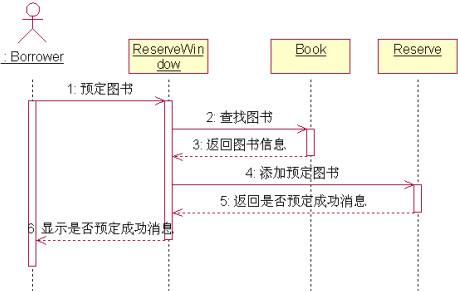
46

应用题



47

应用题



48



协作图

内容

- 1. 什么是协作图
- 2. 组成协作图的元素
- 3. 时序图与协作图的比较
- 4. 常见问题分析
- 5. 使用Rose创建协作图
- 6. 在项目中创建协作图案例分析
- 7. 小结

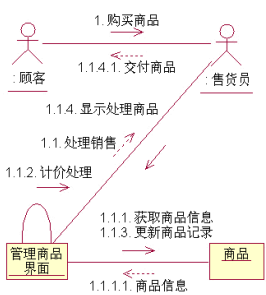
什么是协作图

1. 协作图的基本概念

- 所谓协作是指在一定的语境中一组对象以及用以实现某些行为的这些对象间的相互作用。它描述了在这样一组对象为实现某种目的而组成相互合作的“对象社会”。
- 协作图就是表现对象协作关系的图，它表示了协作中作为各种类元角色的对象所处的位置，在图中主要显示了类元角色（Classifier Roles）和关联角色（Association Roles）。

什么是协作图

- 在UML的表示中，协作图将类元角色表示为类的符号（矩形），将关联角色表现为实线的关联路径，关联路径上带有消息符号。
- 不带有消息的协作图不标明了交互作用发生的上下文，而不表示交互。它可以用来表示单一操作的上下文，甚至可以表示一个或一组类中所有操作的上下文。如果关联线上标有消息，图形就可以表示一个交互。典型的，一个交互用来代表一个操作或者用例的实现。



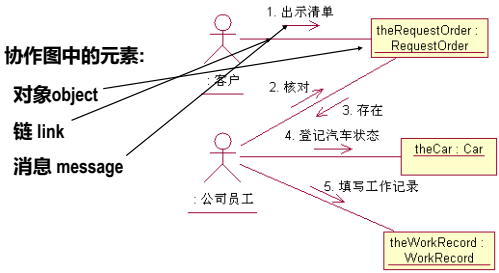
什么是协作图

2. 为什么要使用协作图

- 协作图作为一种在给定语境中描述协作中各个对象之间的组织交互关系的空间组织结构图图形化方式，在使用其进行建模时，可以将其作用分为以下三个方面：
 1. 通过描绘对象之间消息的传递情况来反映具体的使用语境的逻辑表达。一个使用语境的逻辑可能是一个用例的一部分，或是一条控制流。这和序列图的作用类似。
 2. 显示对象及其交互关系的空间组织结构。协作图显示了在交互过程中各个对象之间的组织交互关系以及对象彼此之间的链接。与序列图不同，协作图显示的是对象之间的关系，并不侧重交互的顺序，它没有将时间作为一个单独的维度，而是使用序列号来确定消息及并发线程的顺序。
 3. 协作图的另外一个作用是表现一个类操作的实现。协作图可以说明类操作中使用到的参数、局部变量以及返回值等。当使用协作图表现一个系统行为时，消息编号对应了程序中嵌套调用结构和信号传递过程。

协作图

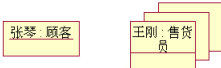
- 协作图的概念：
 - 用于描述系统的行为是如何由系统成分协作实现的图。
- 协作图中的样式和元素



组成协作图的元素

1. 对象

- 一个协作代表了为了完成某个目标而共同工作的一组对象。对象的角色表示一个或一组对象在完成目标的过程中所应起的那部分作用。对象是角色所属的类的直接或者间接实例。
- 在协作图中，不需要关于某个类的所有对象都出现，同一个类的对象在一个协作图中也可能要充当多个角色。
- 协作图中对象的表示形式也和序列图中的对象的表示方式一样，使用包围名称的矩形框来标记，所显示的对象及其类的名称带有下划线，二者用冒号隔开，使用“对象名：类名”的形式，与序列图不同的是，对象的下部没有一条被成为“生命线”的垂直虚线，并且对象存在多对象的形式。



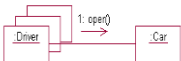
协作图

● 协作图样式和元素

- 主动对象：主动对象是有一方法可以自动启动执行，框加粗。



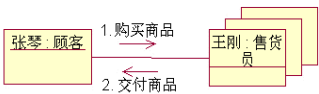
- 多对象：表示同属于一个类的多个对象集合。



组成协作图的元素

2. 消息

- 在协作图中，可以通过一系列的消息来描述系统的动态行为。
- 在协作图中，消息使用带有标签的箭头来表示，它附在连接发送者和接收者的链上。
- 链连接了发送者和接收者，箭头的指向便是接收者。每个消息包括一个顺序号以及消息的名称。
- 消息的名称可以是一个方法，包含一个名字和参数表、可选的返回值表。消息的各种实现的细节也可以被加入，如同步与异步等。



组成协作图的元素

3. 链

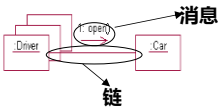
- 在协作图中的链和对象图中链的概念和表示形式都相同，都是两个或多个对象之间的独立连接，是对象引用元组（有序表），是关联的实例。
- 在协作图中，链的表示形式为一个或多个相连的线或弧。在自身相关联的类中，链是两端指向同一对象的回路，是一条弧。为了说明对象是如何与另外一个对象进行连接的，我们还可以在链的两端添加上提供者和服务端的可见性修饰。如下图所示，是链的普通和自身关联的表示形式。



协作图

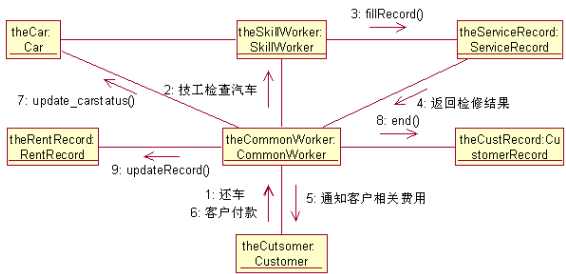
● 协作图样式和元素

- 链和消息：连接对象的线段，以及对对象之间传输的信息。

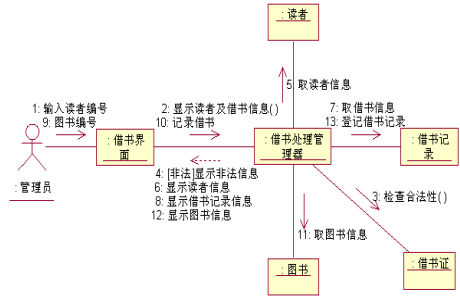


一个链上可以有多个消息。链上可以有角色名、导航、链的两端是否有聚集关系，但没有多重性说明。

汽车租赁系统中“客户还车”的协作图



实例：图书馆借书处理的协作图



时序图与协作图的比较

- 相同点**
- 1.二者均直观地规定了发送对象和接收对象的责任.
 - 2.二者均描述系统中对象之间的动态关系.
 - 3.两种图的语义是等价的, 可以从一种形式的图转换成另一种形式的图, 而不丢失任何信息.
- 不同点**
- 1.协作图强调参与交互的对象的结构组织; 时序图强调消息的时间顺序.
 - 2.时序图可以直观反映对象的创建/激活/撤消; 协作图一般不能.
 - 3.协作图能反应动作路径, 消息必须有顺序号; 时序图没有.

常见问题分析

(1) 如何在时序图中表示消息的循环发送?

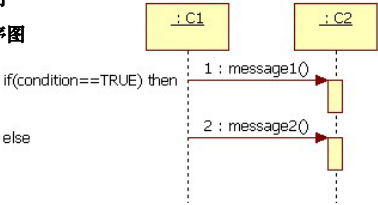
在消名前加循环条件, 例如:

- 1.1 *[for all order lines]: message1()
- 2.1 *[i:= 1..n]: message2()

常见问题分析

(2) 如何在时序图中表示消息的条件发送?

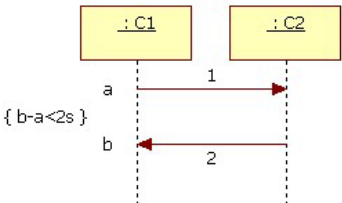
- ① 在消息上加警戒条件, 如: $[x < 0]$ 4: invert(x, color)
- ② 在消息名字前加条件子句, 如: 3.1 $[x > y]$: message()
- ③ 使用文字说明
- ④ 分成多个时序图



常见问题分析

(3) 如何在时序图表示时间约束?

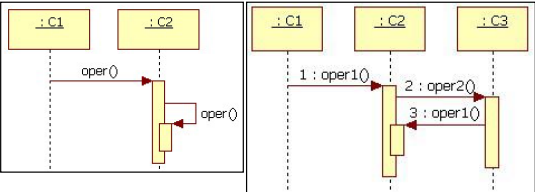
可以使用UML的3种扩展机制之一——约束(constraint)来表示, 如:



常见问题分析

(4) 如何在时序图中表示递归?

直接递归 **间接递归**



常见问题分析

(5) 时序图中对象如何确定？
时序图与类图是否同时生成并交互修改？

时序图中的对象的确定需要参考类图，类图中的类的确定也需要参考时序图，二者相互补充、相互协调的关系。

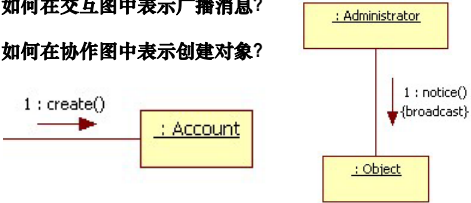
在分析阶段，时序图中的消息名可能只是一个说明，而在设计阶段，时序图的消息名被细化，最后对应到类图中的方法。

常见问题分析

(6) 如果对象具有多态性，发送对象不可能事先知道目标对象所属的类，则在交互图中如何确定目标对象所属的类？
目标对象应是形成多态的继承关系中的祖先类。

(7) 如何在交互图中表示广播消息？

(8) 如何在协作图中表示创建对象？



使用Rose创建协作图

1. 创建对象

- 创建一个新的协作图，可以通过以下方式进行。
 - (1) 右键单击浏览器中的Use Case View (用例视图)、Logical View (逻辑视图) 或者位于这两种视图下的包。
 - (2) 在弹出的菜单中，选中“New”（新建）下的“Collaboration Diagram”（协作图）选项。
 - (3) 输入新的协作图名称。
 - (4) 双击打开浏览器中的协作图。
- 使用菜单栏中添加对象的步骤如下：
 - (1) 使用工具栏时，在菜单栏中，选择“Tools”（浏览）下的“Create”（创建）选项，在“Create”（创建）选项中选择“Object”（对象），此时光标变为“+”号。
 - (2) 在协作图中单击选择任意一个位置，系统在该位置创建一个新的对象。
 - (3) 在对象的名称栏中，输入对象的名称。这时对象的名称也会在对象上端的栏中显示。

使用Rose创建协作图

2. 创建消息

- 在协作图中添加对象与对象之间的简单消息的步骤如下：
 - (1) 选择协作图的图形编辑工具栏中的图标，或者选择菜单栏“Tools”（工具）中“Create”（新建）下的“Message”选项，此时的光标变为“+”符号。
 - (2) 单击连接对象之间的链。
 - (3) 此时在链上出现一个从发送者到接收者的带箭头的线段。
 - (4) 在消息线段上输入消息的文本内容即可。



使用Rose创建协作图

3. 创建链

- 在协作图中创建链的操作与在对象图中创建链的操作相同，可以按照在对象图中创建链的方式进行创建。同样我们也可以在链的规范对话框的“General”选项卡中设置链的名称、关联、角色以及可见性等。
- 链的可见性是指一个对象是否能够对另一个对象可见的机制。

可见性类型 ¹⁾	用途 ²⁾
Unspecified ³⁾	默认设置，对象的可见性没有被设置 ⁴⁾
Field ⁵⁾	提供着是客户的一部分 ⁶⁾
Parameter ⁷⁾	提供着是客户一个或一些操作的参数 ⁸⁾
Local ⁹⁾	提供着对客户来讲是一个本地声明对象 ¹⁰⁾
Global ¹¹⁾	提供着对客户来讲是一个全局对象 ¹²⁾

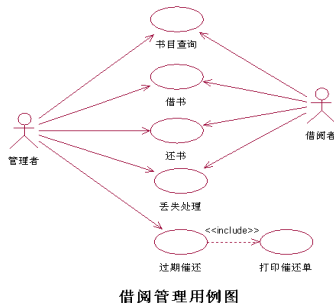
建立协作图的步骤

1. 确定用例中的交互过程
2. 识别参与交互过程的对象
3. 如果需要，为每个对象设置初始特性
4. 确定对象之间的链，及沿着链的消息
5. 从引发该交互过程的初始消息开始，将每个消息附到相应的链上
6. 如果需要说明时间约束，则在消息旁边加上约束说明
7. 如果需要，可以为每个消息设置前置条件和后置条件

思考

绘制图书馆还书处理的时序图和协作图

图书馆的借阅管理用例图



图书馆的还书用例

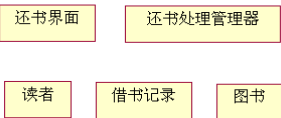
- 用例：还书
- 参与者：管理员, 借阅者
- 操作流：
 - ① 管理员进入图书借阅界面，用例开始。
 - ② 系统要求输入所还图书的条码。
 - ③ 系统显示所还图书的图书名、读者、借阅等信息。
 - ④ 确认还书。
 - ⑤ 系统回到上一界面，等待处理下一业务。

思考：图书馆还书处理的时序图和协作图

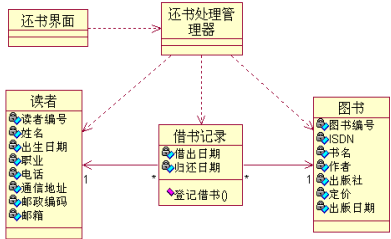
- 识别交互过程
 - 读者在还书时，先由管理员把所借图书的图书编号扫描给系统，系统接收到这个信息，则显示这个读者信息，以及这本书的信息。
 - 管理员确认还书，则系统登记还书信息，并返回还书成功信息，还书过程完成。

思考：图书馆还书处理的时序图和协作图

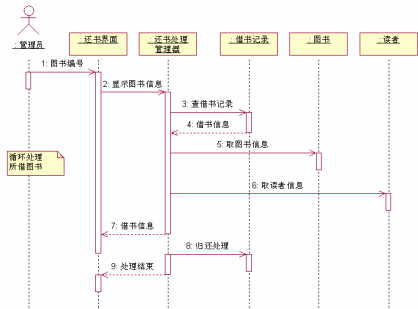
- 识别参与交互过程的对象



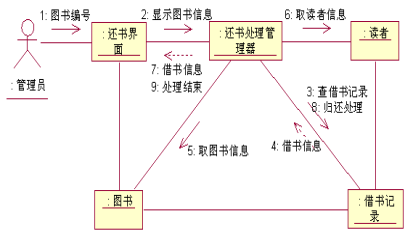
思考：图书馆还书处理的时序图和协作图



图书馆还书处理的时序图



图书馆还书处理的协作图

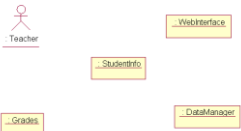


在项目中创建协作图案例分析

1. 需求分析
- 我们可以通过更加具体的描述来确定工作流程，基本工作流程如下：
(1) 李老师希望通过系统查询某名学生的学科成绩。
(2) 李老师通过用户界面录入学生的学号以及学科科目请求学生信息。
(3) 用户界面根据学生的学号向数据库访问层请求学生信息。
(4) 数据库访问层根据学生的学号加载学生信息。
(5) 数据库访问层根据学生信息和学科科目获取该名学生的分数信息。
(6) 数据库访问层将学生信息和分数信息提供给用户界面。
(7) 用户界面将学生信息和分数信息显示出来。

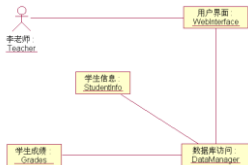
在项目中创建协作图案例分析

2. 确定协作图元素
- 从已经描述的用例中，我们可以确定需要“教师”、“学生”和“成绩”对象，我们还需要一个提供教师与系统交互的场所，那么我们需要一个“用户界面”对象。“用户界面”对象如果要获取“学生”和“成绩”对象的信息，那么我们还需要一个用来访问数据库的对象。将这些对象列举到协作图中。



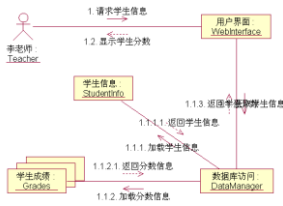
在项目中创建协作图案例分析

3. 确定元素间的关系
- 创建协作图的下一步是确定这些对象之间的连接关系，使用链和角色将这些对象连接起来。在这一步中，我们基本上可以建立早期的协作图，表达出协作图中的元素如何在空间上进行交互。



在项目中创建协作图案例分析

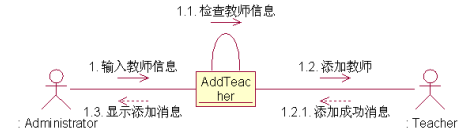
4. 完成协作图



练习题

(1) 以“远程网络教学系统”为例，在该系统中，系统管理员需要登录系统才能进行系统维护工作，如添加教师信息、删除教师信息等。根据系统管理员添加教师信息用例，创建相关协作图。

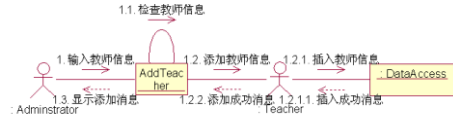
练习题



练习题

(2) 在“远程网络教学系统”中，如果我们单独抽象出来一个数据访问类来进行数据访问。那么，根据系统管理员添加教师信息用例，重新创建相关协作图。

练习题



小结

- 交互图适用于条件判断和循环不太多的时序过程。
- 主要在一个用例中，描述对象协同工作的行为。
- 描述跨越多个用例的单个对象的行为，应当使用状态图。
- 描述跨越多个用例或多线程的复杂行为，应当使用活动图。

选择题

- 时序图由(), 生命线, 激活期和消息组成
 - A. 关系
 - B. 类角色
 - C. 用例
 - D. 实体

• 选B

选择题

- 多对象是UML哪个视图中的概念 ()
A. 类图
B. 状态图
C. 协作图
D. 组件图

● 选C

91

选择题

- 下列描述中，不能正确描述协作图的是 ()
A. 协作图作为一种交互图，强调的是参加交互的对象组织；
B. 协作图是时序图的一种特例
C. 协作图中有消息流的时序号；
D. 在ROSE工具中，协作图可在时序图的基础上按“F5”键自动生成；

● 选B

92

填空题

- UML提供一系列的图支持面向对象的分析与设计，其中时序图和协作图都是描述系统动态视图的交互图，_____描述了以时间顺序组织的对象之间的交互活动。

● 时序图

93

简答题

- 试比较UML中的时序图和协作图
- 时序图和协作图都可可视化地展示对象之间的交互关系。
(1分) 时序图强调的是交互的时间时序，而协作图强调的是交互的语境和参与交互的对象的整体组织。(1分) 时序图按照时间顺序布图，而协作图按照空间组织布图。(1分) 时序图可以清晰地表示消息之间的顺序和时间关系，但需要较多的水平方向的空间。(1分) 协作图在增加对象时比较容易，而且分支也比较少，但如果消息比较多时难以表示消息之间的顺序。(1分)

94