

第三章

使用Visio绘制时序图



本章目标

□掌握利用Visio绘制程序时序图基本操作



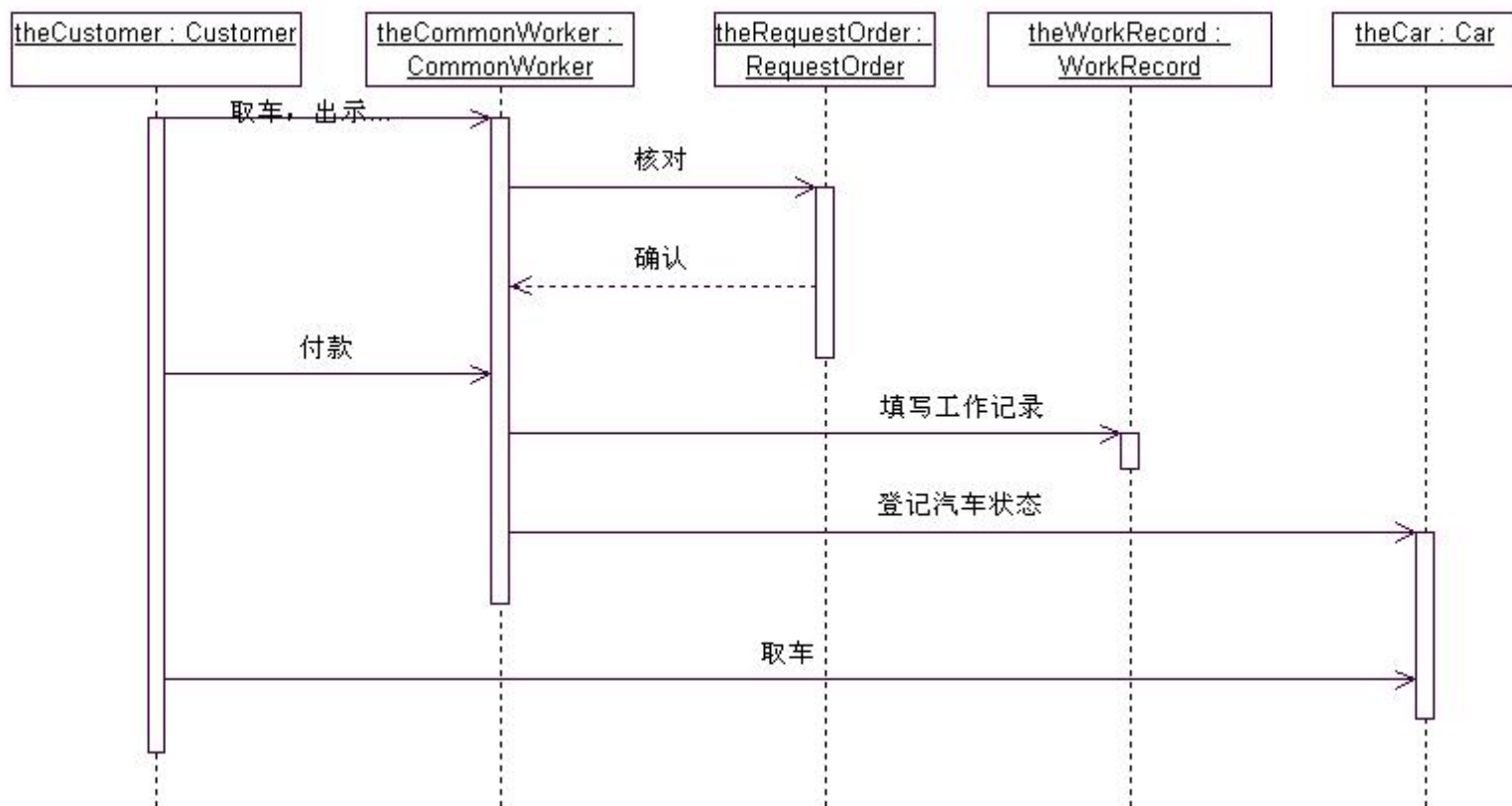
- 为什么要绘制时序图？
- 概述
- 时序图的组成
- 对象的创建和撤销
- 时序图建模技术



□我们编码的时候，知道有的用例的业务逻辑按照比较确定的时间先后顺序进行展开。这时候，我们就需要知道我们设计的系统中的不同类之间传递消息（可以认为是不同对象函数间的调用）要按照怎么样的顺序、传递什么消息、返回什么消息。这时候用时序图是最好不过的了。

- 时序图是强调消息时间顺序的交互图。
- 时序图描述了对象之间传送消息的时间顺序，用来表示用例中的行为顺序。
- 时序图将交互关系表示为一个二维图。其中，纵轴是时间轴，时间沿竖线向下延伸。横轴代表了在协作中各独立的对象。

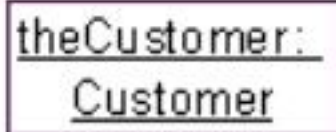




- ❑ 对象 (Object)
- ❑ 生命线 (Lifeline)
- ❑ 消息 (Message)
- ❑ 激活 (Activation)



□符号:

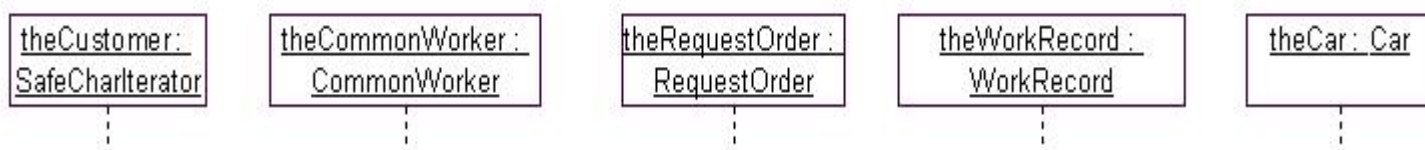


```
theCustomer:  
Customer
```

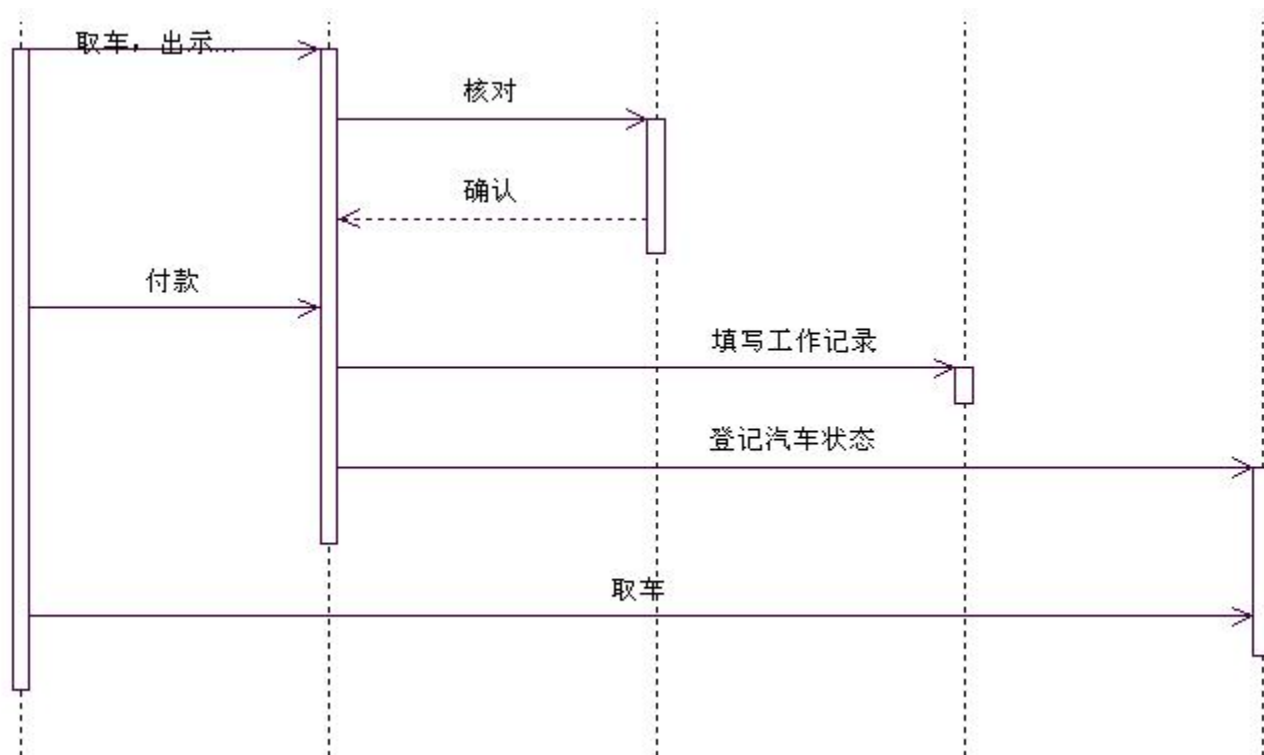
□将对象置于时序图的顶部意味着在交互开始的时候对象就已经存在了，如果对象的位置不在顶部，那么表示对象是在交互的过程中被创建的。

- ❑ 生命线是一条垂直的虚线，表示时序图中的对象在一段时间内的存在。每个对象的底部中心的位置都带有生命线。
- ❑ 生命线是一个时间线，从时序图的顶部一直延伸到底部，所用的时间取决于交互持续的时间。
- ❑ 对象与生命线结合在一起称为对象的生命线，对象的生命线包含矩形的对象图标以及图标下面的生命线。





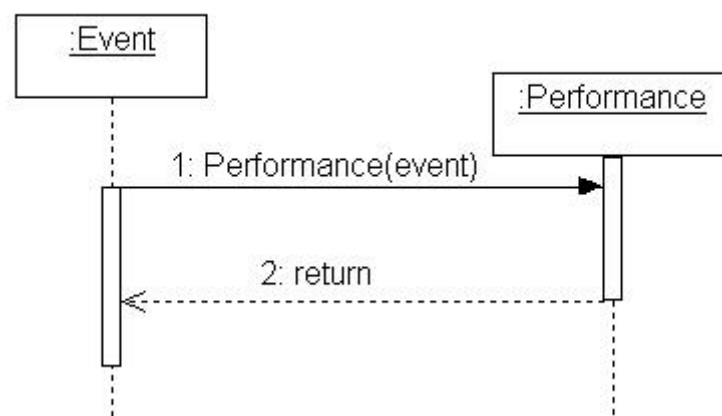
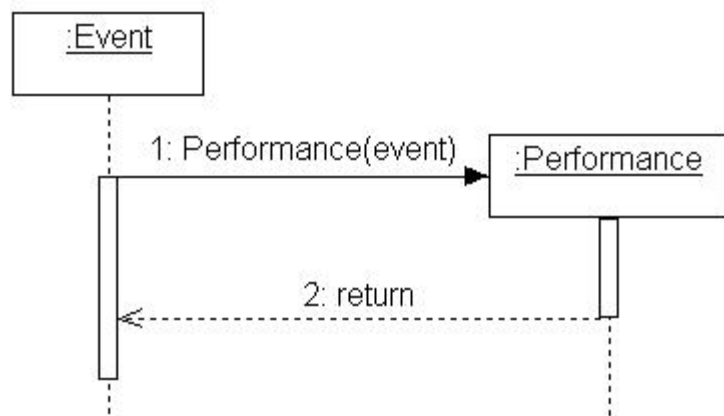
- ❑ 消息定义的是对象之间某种形式的通信，它可以激发某个操作、唤起信号或导致目标对象的创建或撤销。
- ❑ 消息是两个对象之间的单路通信，从发送方到接收方的控制信息流。
- ❑ 消息可以用于在对象间传递参数。
- ❑ 消息可以是信号，也可以是调用。
- ❑ 在UML中，消息使用箭头来表示，箭头的类型表示了消息的类型。



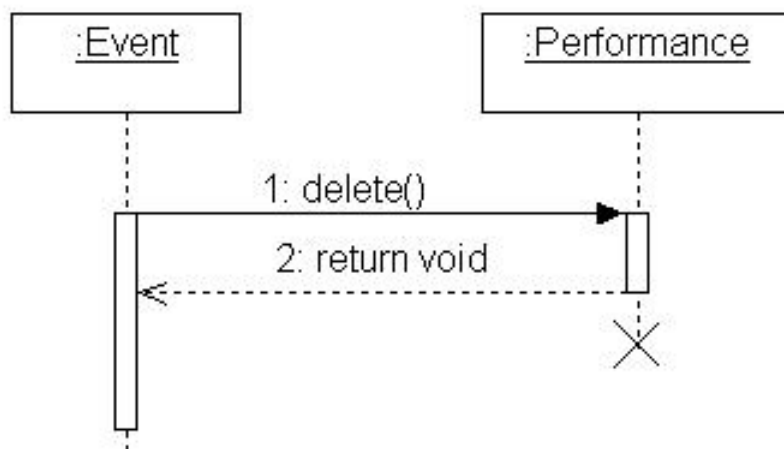
- ❑ 如果对象位于时序图的顶部，说明在交互开始之前该对象已经存在了。如果对象是在交互的过程中创建的，那么它应当位于图的中间部分。
- ❑ 对象在创建消息发生之后才能存在，对象的生命线也是在创建消息之后才存在的。



◆ 创建对象的两种表示方法



□ 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。

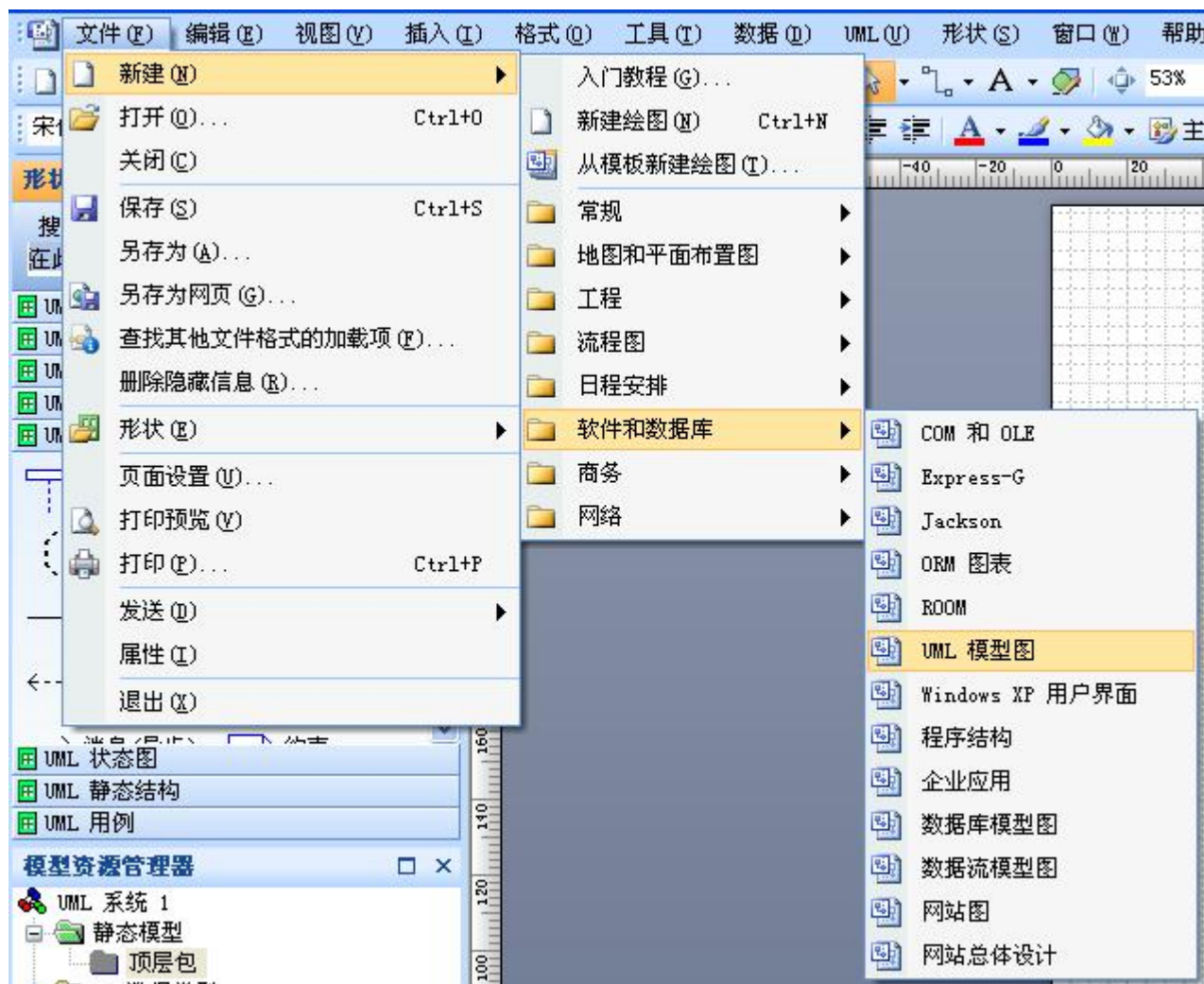


- 设置交互的语境。
- 通过识别对象在交互中扮演的角色，设置交互的场景。
- 为每个对象设置生命线。
- 从引发某个消息的信息开始，在生命线之间画出从顶到底依次展开的消息，显示每个消息的特性（如参数）。
- 如果需要可视化消息的嵌套或实际计算发生时的时间点，可以用激活修饰每个对象的生命期。

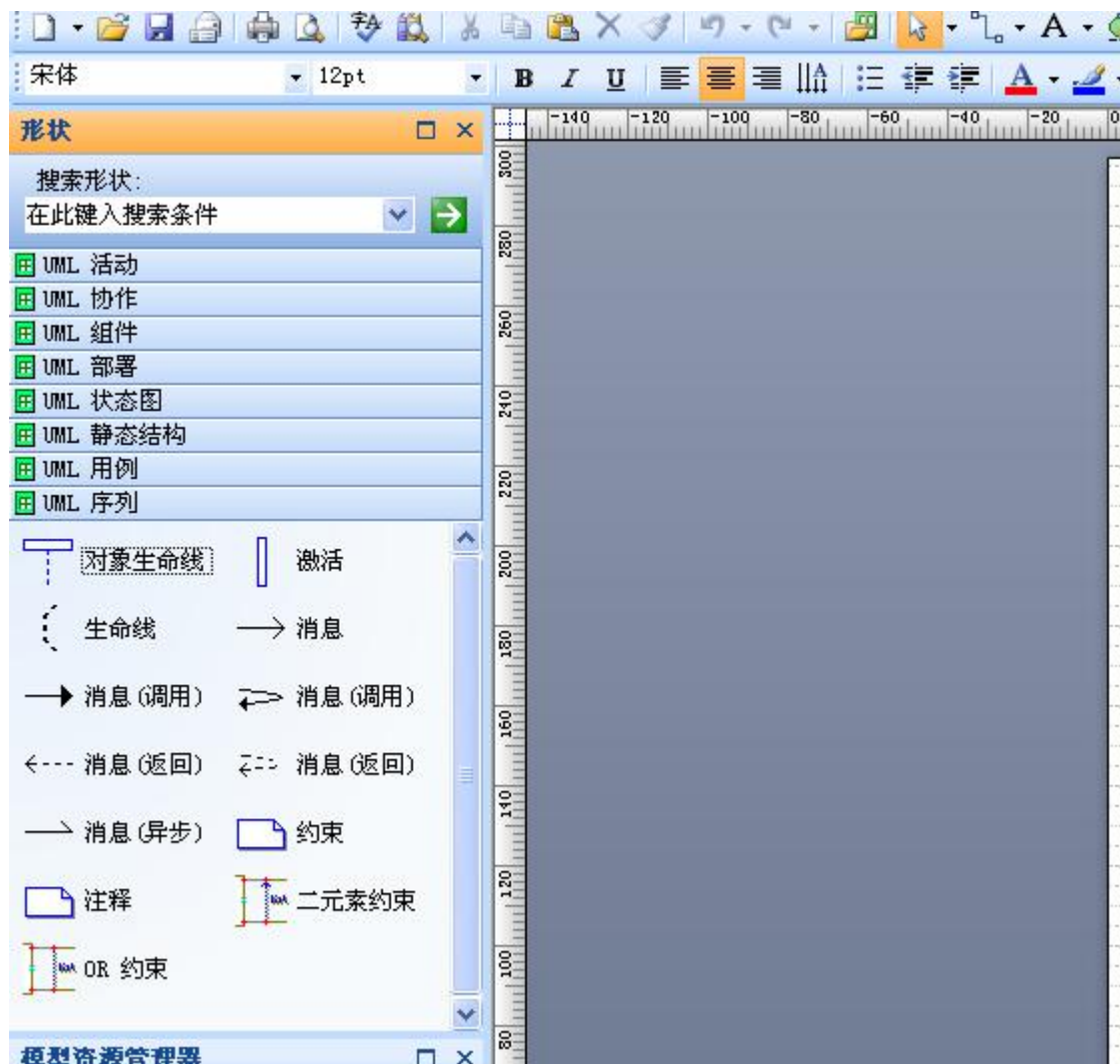
- 如果需要说明时间或空间的约束，可以用时间标记修饰每个消息，并附上合适的时间和空间约束。
- 如果需要更形式化的说明某控制流，可以为每个消息附上前置和后置条件。



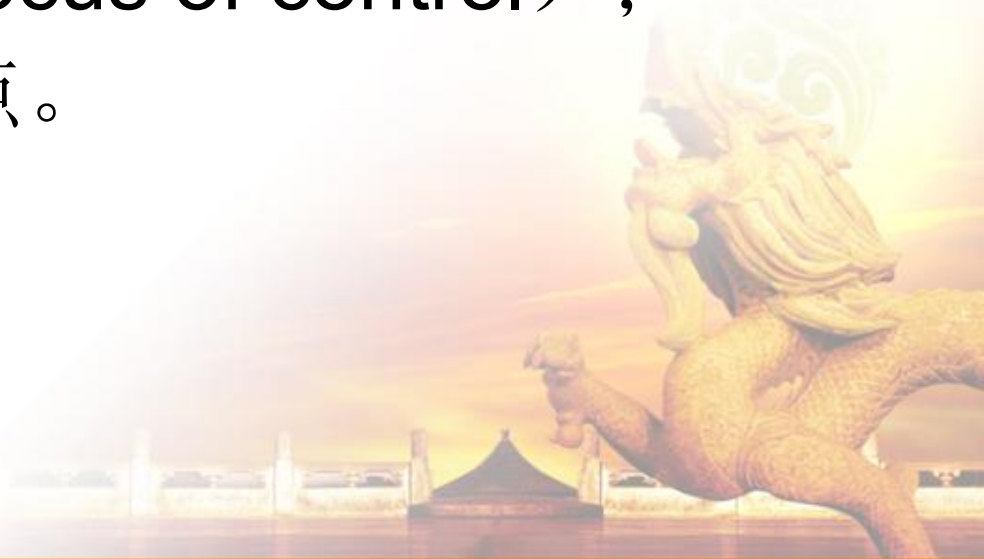
□ 选择模版



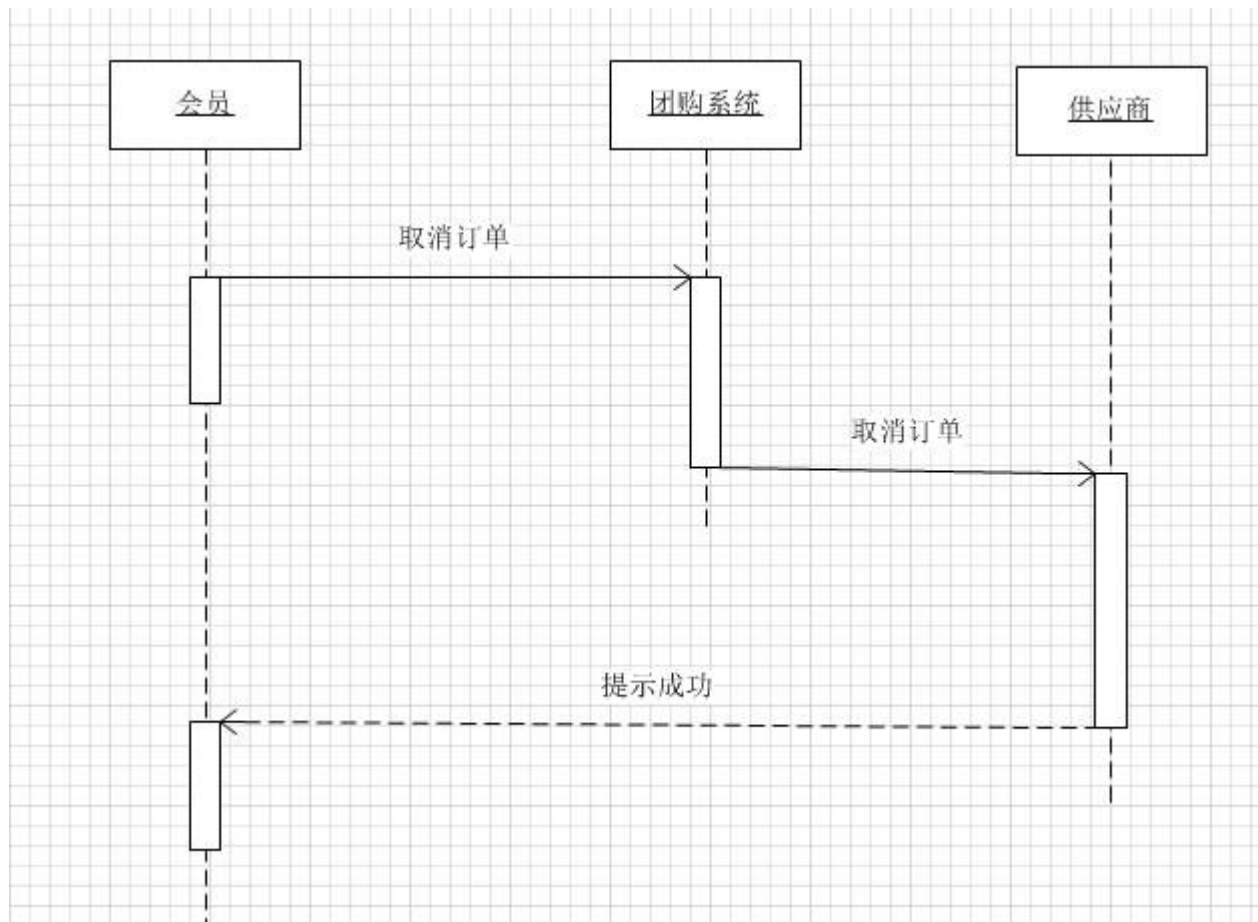
□ 常见符号



- ❑ 确定交互过程的上下文；
- ❑ 识别参与过程的交互对象；
- ❑ 为每个对象设置生命线；
- ❑ 从初始消息开始，依次画出随后消息；
- ❑ 考虑消息的嵌套，标示消息发生时的时间点，则采用FOC（focus of control）；
- ❑ 说明时间约束的地点。



□ 团购系统的取消订单



- 添加用户：（管理员）打开添加用户的界面，在界面上选择一个组（可以通过一个下拉选择框来选择），然后向后台提交，后台系统保存用户的信息（并同时建立用户和组之间的关联）

