
数据库原理及应用

——数据库设计需求分析

内容安排

- 参与者识别
- 用例识别
- 用例图画法

用例图要素

- 用例图显示谁将是相关的用户、用户希望系统提供什么服务以及用户需要为系统提供的服务。

元素：

- ①参与者 (Actor)
- ②用例 (Use Case)
- ③关系：

参与者与用例间：关联关系 (Association)

用例与用例间：包含关系 (Include)

扩展关系 (Extend)

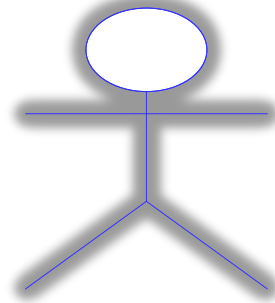
泛化关系 (Generalization)

1 识别参与者

- 参与者，Actor

- 关键词：边界

- 参与者：在系统之外，透过系统边界与系统进行有意义交互的任何事物



思考：识别参与者？

- 天气预报系统：用户如果预定了天气预报，系统每天定时给他发天气消息；如果当天气温高于35度，还要提醒用户注意防暑；

在这个叙述里，谁是Actor？

用户！

气温！

时间！

2 识别用例

- **关键词：价值**

- **定义**

- 用例是**系统执行的一系列动作**，**这些动作将生成特定参与者可观测的结果值**

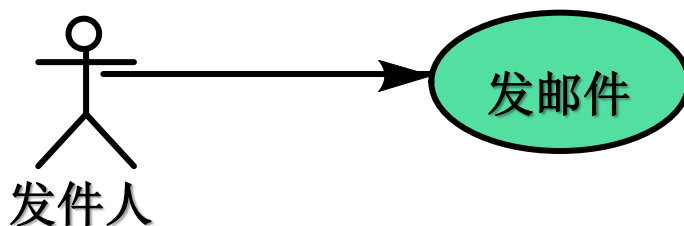
- **简洁：参与者使用系统达到目标**



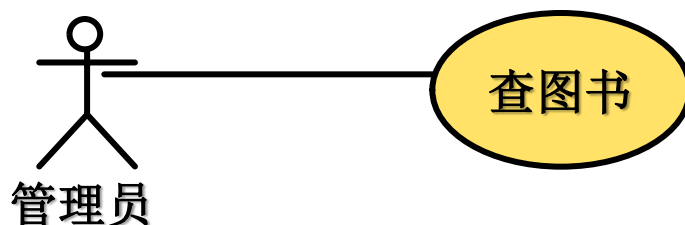
发邮件

关联关系

- 在用例图中，通常存在两种类型的关联：
 - 单向关联

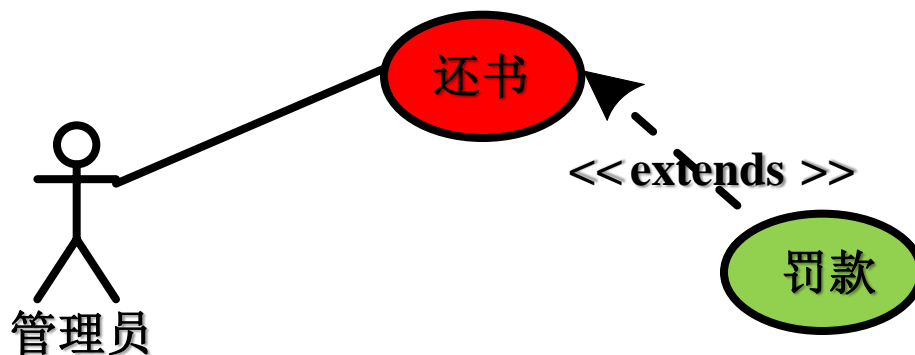


- 双向关联



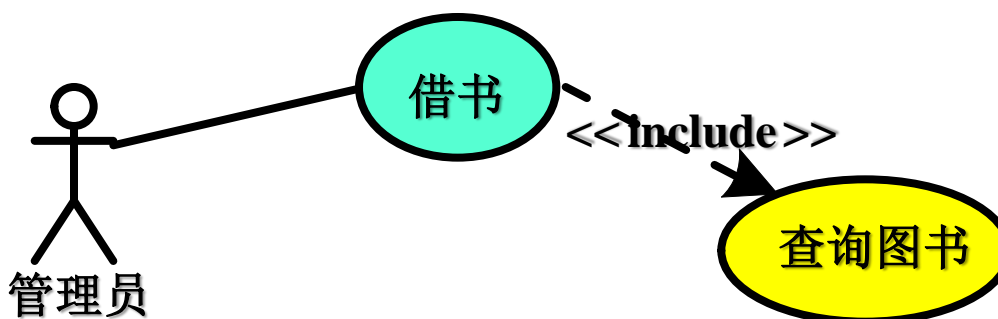
扩展关系

- 基用例路径本身是完整的，可能是一条扩展路径。
 - 基用例在扩展点激活扩展用例，扩展用例执行完，回到基用例扩展点，也就是说，基用例本身是完整的，不扩展也行的通，扩展是其一选做。



包含关系

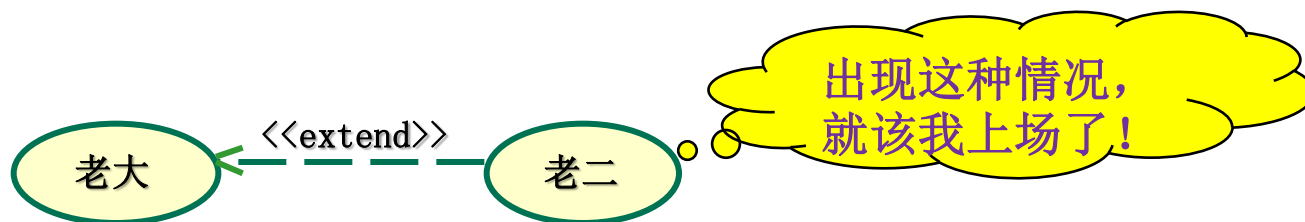
- 某些步骤在多个用例重复出现，且单独形成价值
- 包含关系，基用例路径本身是不完整的，包含用例是基用例的一个步骤。



扩展 VS 包含



老大知道老二

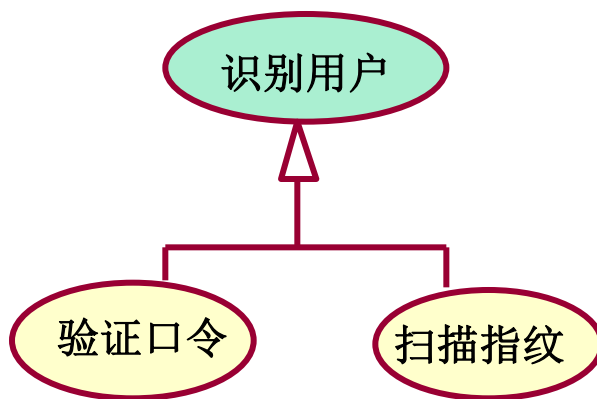


老二知道老大

泛化关系

- 同一业务 **目的不同技术实现** 实现：

一个用例可以特化另一个更普通用例（更普通用例泛化特殊用例）

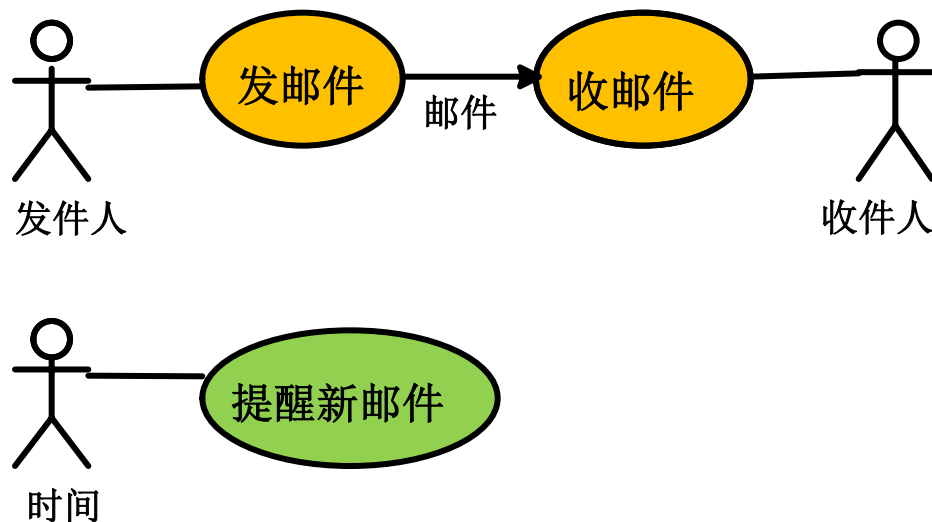


用例图画法

- ①识别参与者
- ②寻找用例
- ③确定关系

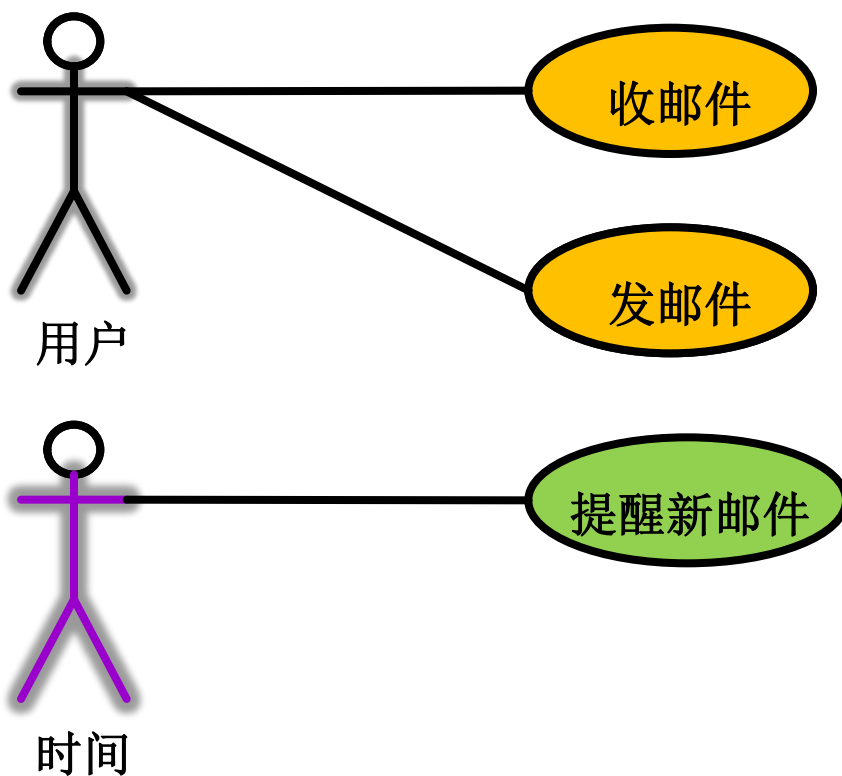
思考：

- **Email**客户端（如：**outlook express**），**A**在北京发邮件给武汉的**B**，系统提醒**B**你有“新邮件”，**B**收邮件



用例是一个完整的交互，用例之间没有顺序的关系

思考答案：

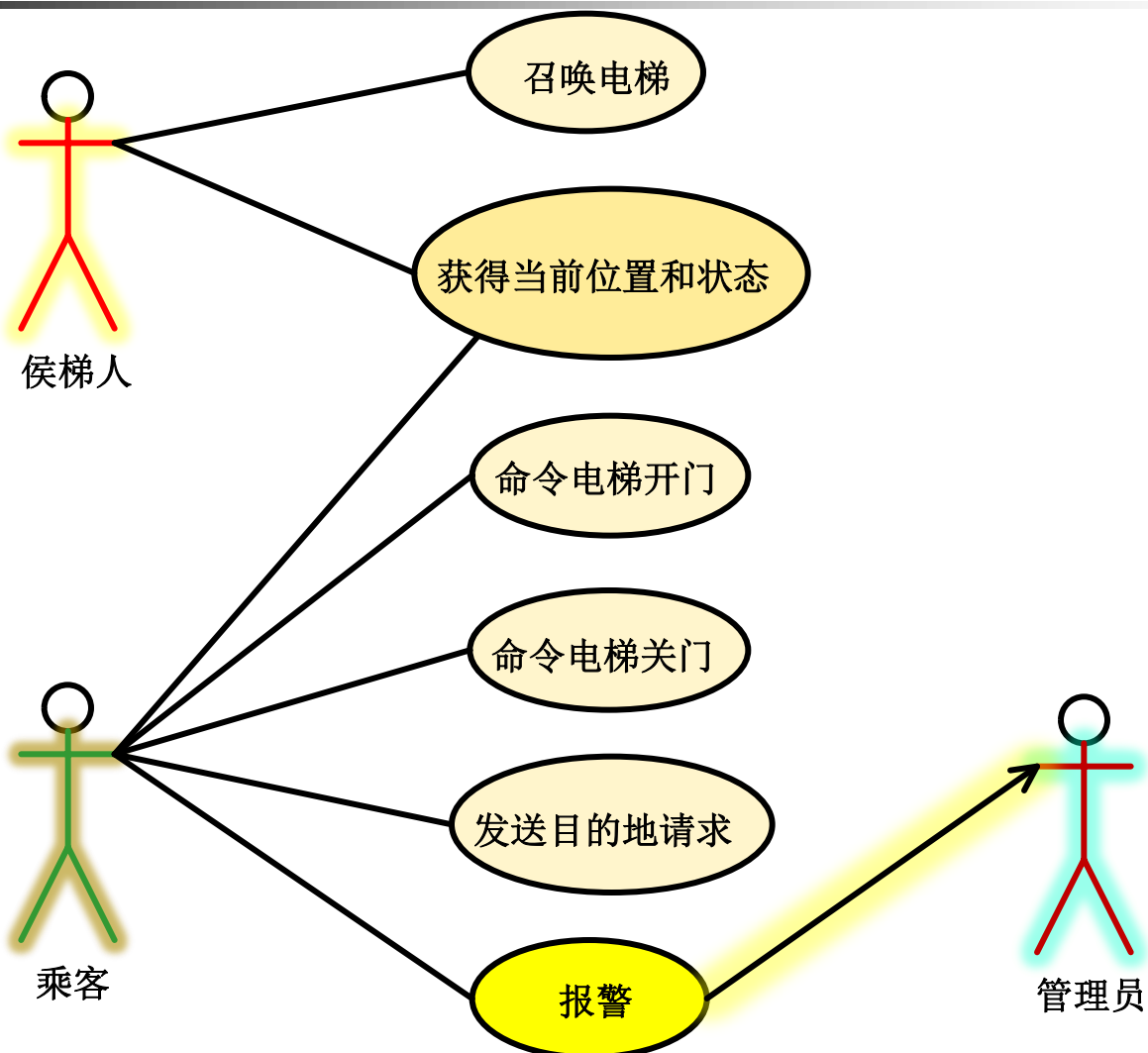


识别用例：讨论和练习



电梯——参与者和用例？

电梯用例图



总结

用例图

步骤

- 识别参与者
- 识别用例
- 确定联系



谢谢!!!



www.hesee.com