



软件工程

Software Engineering

廖清

liaoqing@hit.edu.cn

计算机科学与技术学院



课程简介

- 任课老师：
 - 廖清，
 - C栋303B室
 - liaoqing@hit.edu.cn
 - 0755-86134382
- 课程网站
 - 课件、作业要求、各类通知/消息均在此网站发布：liaoqing.me
- 教材及参考书
 - Rogers S. Pressman. Software Engineering: A Practitioner's Approach (Seventh Edition). (《软件工程：实践者的研究方法，第8版》，机械工业出版社，2016年11月，ISBN 9787111548973)；
 - 孙家广、刘强，《软件工程-理论、方法与实践》，高等教育出版社，2010年11月，ISBN 978-7-040-16308-7；
- 考试
 - 平时作业10%、实践项目40%、期末考试50%



善用开源社区

助教

- 韩浩: hanhao0125@foxmail.com ,C栋303C室, 18688796843
- 杨林: yanglincs17@foxmail.com ,C栋303C室, 13580545033
 - Github上有注册用户2700万人
 - GitHub托管2000多万个开源软件, StackOverflow超过8600万条在线知识条目





教学目标

课程目标：从工程学角度认识软件在大型系统中的设计和应用，具备作为软件工程师从事工程实践所需的专业能力

能力培养目标：

- 需求定义与分析
- 权衡和选择设计方案
- 使用UML建模
- 开发高质量软件
- 运用软件工程工具
- 团队协作开发
- 谈判与沟通

转变对软件开发的认识：程序  系统

转变思维定式：程序员  软件工程师



实践项目

- 16学时，实验课上报告+课后实践。
- 学生每4-5人一组，选定一个来自实际需求的应用软件系统，利用所学软件工程知识进行需求分析、设计、开发、测试、过程与项目管理、源代码管理、版本控制。
- 期间组织4次口试(立项、需求分析、设计与开发、最终验收)。
- 不管是4人一组、5人一组、独立完成，组内成员均获得同样的成绩。

实践项目	40%	开题报告	10%
		第一轮迭代	10%
		第二轮迭代	10%
		测试验收	10%



课程章节安排

- 第一章 软件工程概论
- 第二章 软件项目开发过程与管理
- 第三章 软件需求工程
- 第四章 软件设计
- 第五章 软件编码、测试与质量保障
- 第六章 软件实施、维护与演化
- 第七章 软件工程新技术展望
- 第八章 案例分析
- 实践项目

理论课：32学时

实践课：16学时



本章內容

- 用户故事 (User Story) 案例
- 用例图 (User Case Diagram)
- 活动图 (Activity Diagram)



User Story Card Example

• Card [优先级: xx 工作量估算:]

➤ 作为一个视频制作者，我希望上传视频到网页，以便于任何其他用户能够在网页上看到这个视频。

• Conversation

- “上传按钮”会出现在这个网站的每一个网页上
- 视频不能大于100MB或者长于10分钟
- 视频的格式包括.flv,.mov,.mp4,.avi和.mpg
- 能够实时显示上传进度



User Story Card Example

- Confirmation

- ✓ 检查“上传”按钮

- ✓ 上传视频的具体操作

1. 检查.flv,.mov,.mp4,.avi和.mpg视频文件是否能上传

2. 检查其它视频文件是否不能上传

3. 检查视频大于100MB时是否会报错

4. 检查视频长于10分钟时是否会报错

- ✓ 检查“上传视频”按钮

- ✓ 检查是否能够实时显示上传进度



常见的User Story错误

- 对用户进行分类，倾听每种用户的需求
 1. 注册用户，非注册用户（游客），开发人员
 2. 注册用户可以看报表，产生报表
 3. 非注册用户（游客）可以看报表，但不能产生报表
 4. 注册用户可以选择加载从xx年-xx年的数据进系统
 5. 非注册用户（游客）不能加载数据，只能使用系统中默认加载的数据
 6. 开发人员可以选择系统支持哪个系统平台，而其他人员不能选择
 7. 开发人员可以选择爬虫爬取xx年-xx年数据
 8. 非开发人员不能操作爬虫，只能选择系统加载xx年-xx年数据
 9. 开发人员还可以再继续细分。。。



常见的User Story错误

- 给出来的user story没有价值/好处
- 例子：作为一个广告商，我希望有一个过滤选项
 - 作为一个广告商，他为什么想要一个过滤选项？他的目的是什么？
- 例子：作为一个商业用户，我希望ATM能够输入密码
 - 作为一个商业用户，他为什么希望ATM能够输入密码？
- 如果没有接受标准或满意条件，可能会使故事由于缺乏理解而失败，后续的测试或者测试用例也将被改变。



课后作业

- 请基于当前你们要完成的项目的实际需求，分别针对不同的客户编写两个用户故事卡片，这两个用户故事卡片里所有内容都会在你们最终的项目中实现。



本章內容

- 用戶故事 (User Story) 案例
- 用例圖 (User Case Diagram)
- 活動圖 (Activity Diagram)



UML-User Case

UML语言-面向对象的建模语言

➤借助**面向对象**的概念和机制来表示系统

- 系统：软件系统，现实系统
- 需要表示机制和语言

➤1980s-1990s出现了大量**面向对象 (Object Oriented)** 建模语言

- 多达几十种
- 代表性有Booch方法、OMT方法和OOSE方法

➤期望

- 各有千秋，却又有很多类似之处，往往让使用者无所适从，需要吸收各种技术的优点
- 寻求一种概念清晰、表达能力丰富、适用范围广泛的面向对象的建模语言



Booch



Rumbaugh



Jacobson



UML-User Case

UML: Unified Modeling Language

➤ Unified(统一)

- 提取不同方法中的最好建模技术，如OMT (James Rumbaugh)，Booch method (Grady Booch) 和OOSE (Ivar Jacobson)
- 统一表示方式，具有标准化的表示手段

➤ Modeling(建模)

- 对现实系统和软件系统进行可视化描述
- 建立系统模型

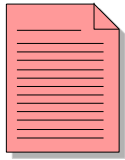
➤ Language(语言)

- 图形语言：语法、语义和语用
- 包括规则，约束扩展机制



UML-User Case

- 用来可视化 (visualize) 、描述 (specify)、构造 (construct) 和文档化 (document) 软件密集型系统的各种产品
- 支持不同人员之间的交流 (Communication)



Specifying



Visualizing



Constructing



Documenting



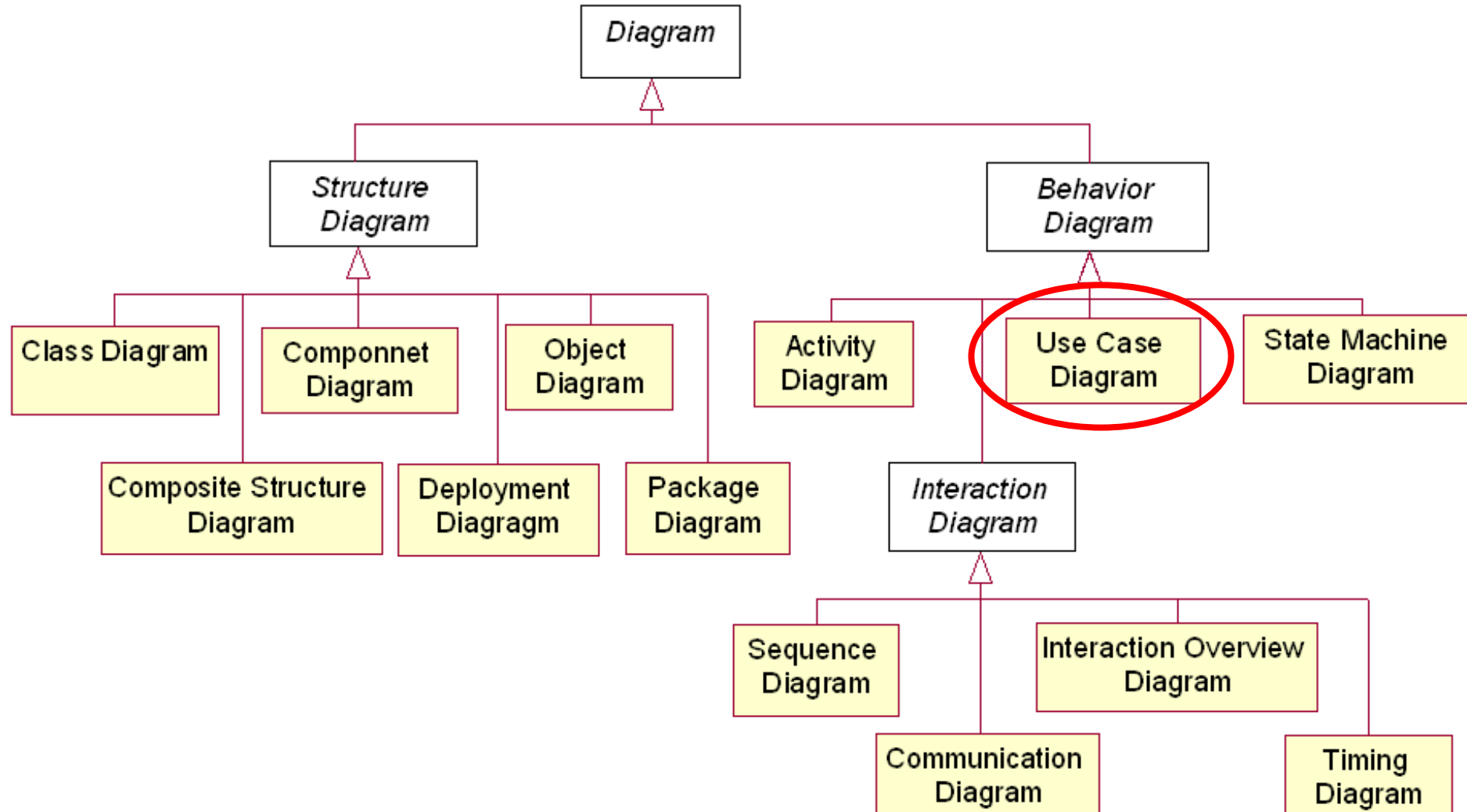
Business Modeling



Communications



UML2.0的图的分类





基本概念

- **参与者(actor)**: 是某些具有行为的事物, 可以是人 (由角色标识)、计算机系统或者组织, 例如收银员
- **场景(scenario)**: 是参与者和系统之间的一系列特定的活动交互, 也称为**用例实例(use case instance)**
- **用例(use case)**: 就是一组相关的成功和失败场景集合, 用来描述参与者如何使用系统来实现其目标

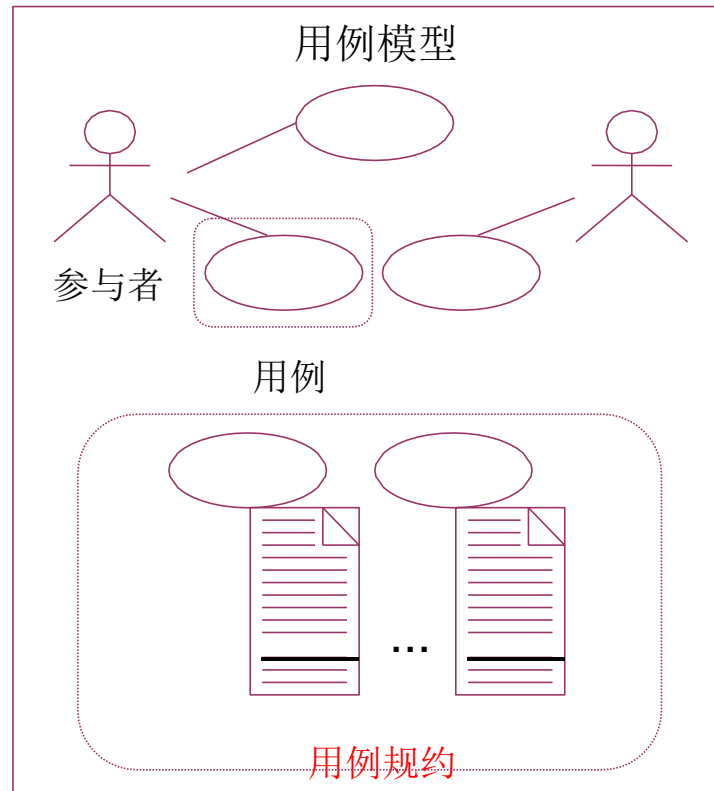


如何绘制用例模型

- **Step 1:** 确定系统边界
- **Step 2:** 识别并描述参与者(actor);
- **Step 3:** 确定每个参与者目标, 识别用例(use case) ;
- **Step 4:** 识别参与者与用例之间的通讯关联(Association);
- **Step 5:** 给出每一个用例的详细描述
- **Step 6:** 细化用例模型

示例：ATM系统的用例

- 参与者：银行客户
- 用例：银行客户使用自动提款机来进行银行帐户的存款、取款和转帐交易

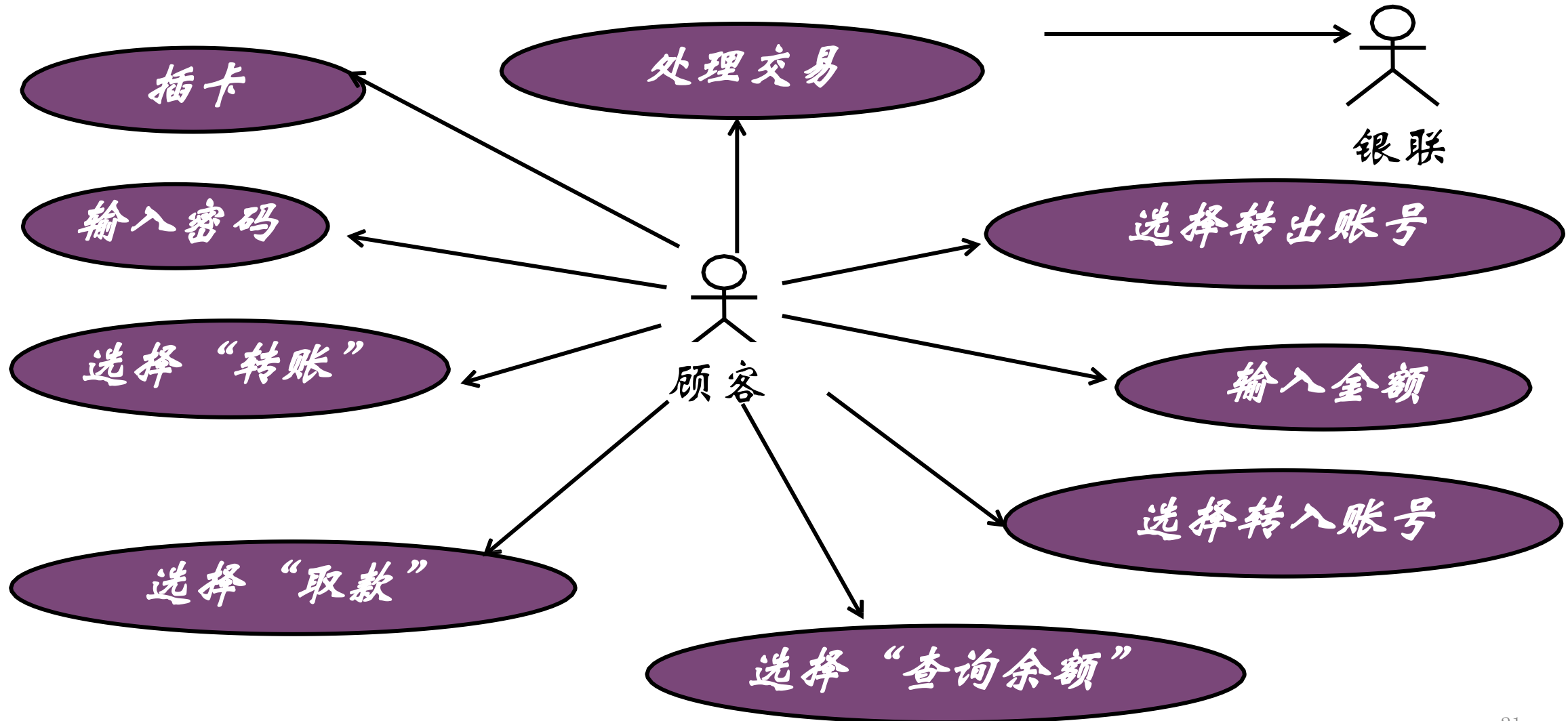


用例名：
参与者及关注点：
主成功场景：
 事件1
 事件2

扩展
前置条件：
后置条件：



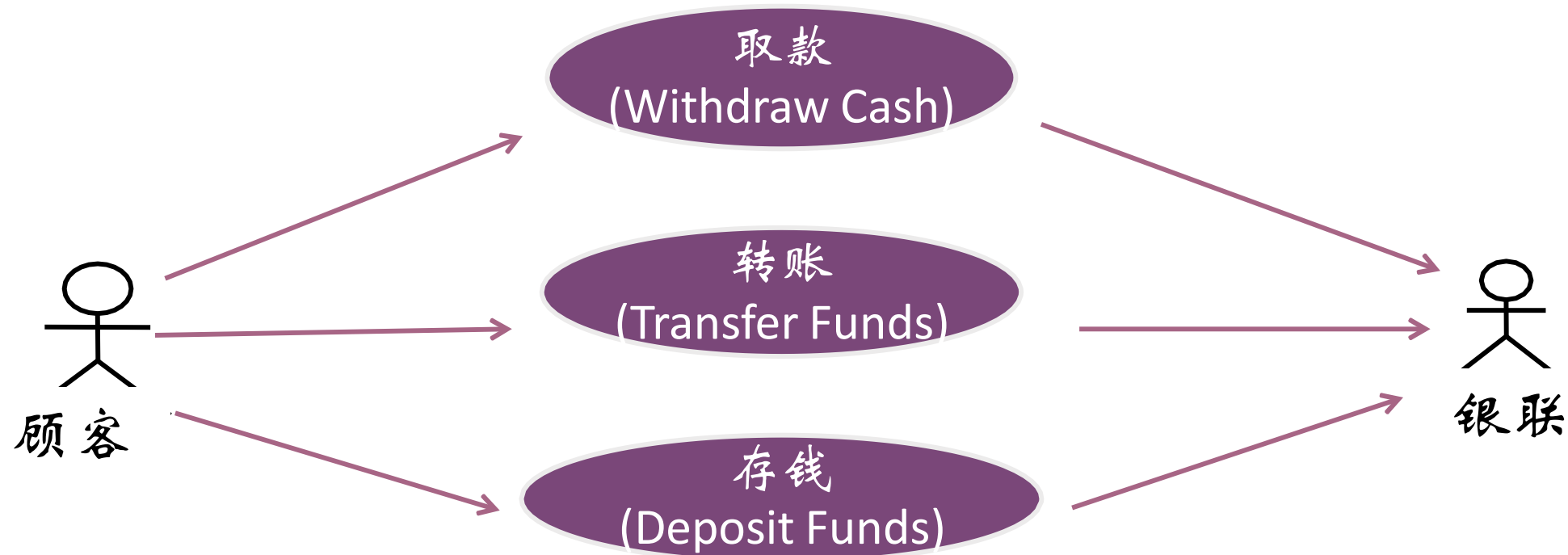
用例常见错误-功能分解





用例常见错误-功能分解

走出功能分解：正确的用例建模





给出每一个用例的详细描述

编写用例文本的准则

- 以无用户界面约束的本质风格编写用例
- 编写简洁的用例
- 编写黑盒用例
- 采用参与者和参与者目标的视角

用例名：
参与者及关注点：
主成功场景：
 事件1
 事件2

扩展
前置条件：
后置条件：
.....



事件流（场景）

退货处理

主要成功场景（常规流）：

- 客户带着要退的货物到达收款处，出纳员使用POS系统记录每一个要退货的货物，...

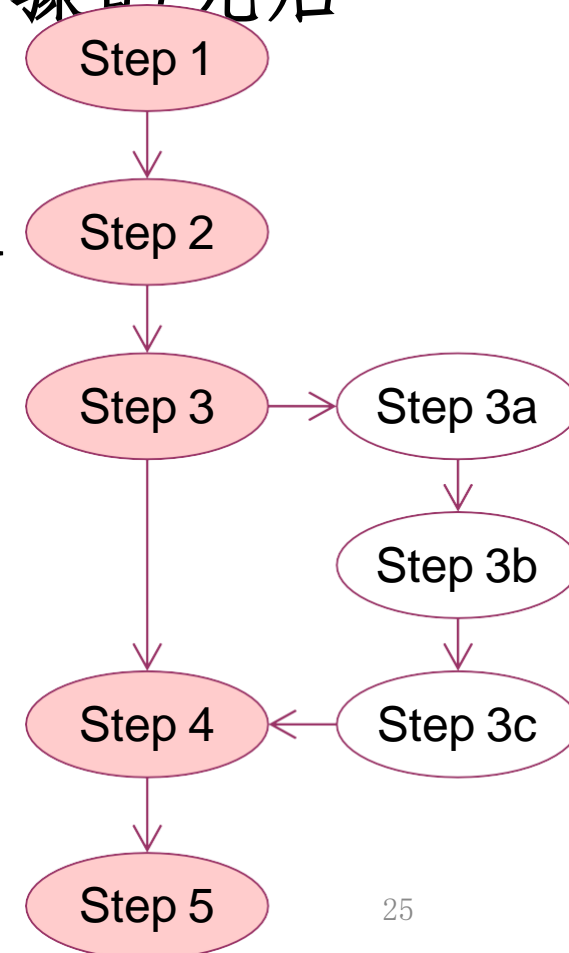
候选场景（扩展流）：

- 若信用验证失败，通知客户并要求使用其他付款方法
- 若系统检测到与外界计税系统通信失败，...



常规事件流（场景）

- 每一个步骤都需要用数字编号以清楚地标明步骤的先后顺序
- 用一句简短的标题来概括每一步骤的主要内容
- 对每一步骤，从正反两个方面来描述
 - 参与者向系统提交了什么信息
 - 对此系统有什么样的响应

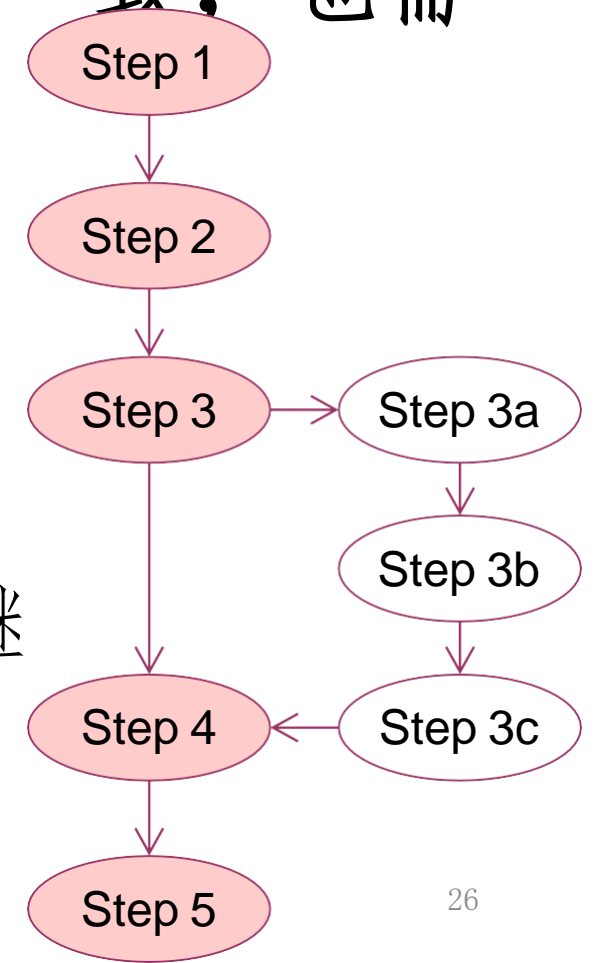




扩展事件流（场景）

■ 扩展流的描述格式可以与基本流的格式一致，也需要编号并以标题概述其内容。

- 起点：该扩展流从事件流的哪一步开始；
- 条件：在什么条件下会触发该扩展流；
- 动作：系统在该扩展流下会采取哪些动作；
- 恢复：该扩展流结束之后，该用例应如何继续执行。





例1

用例：登记借书

1. 目标：

本用例允许图书管理员登记普通读者的借书记录

2 事件流：

1. 常规流程

当读者希望借书、图书管理员准备登记有关的借书记录时，本用例开始执行。

- (1) 系统要求管理员输入读者的注册号和所借图书号；
- (2) 图书管理员输入信息后，系统产生一个唯一的借书记录号；
- (3) 系统显示新生成的借书记录；
- (4) 图书管理员确认后，系统增加一个新的借书记录

2. 扩展流程

(1) 读者没有注册

在主流程中，如果系统没有读者的注册信息，系统将显示错误信息，用例结束；

(2) 所借图书不存在

在主流程中，如果所借图书已被借出或者系统中无该图书，系统将显示错误信息，用例结束。

3 前置条件：用例开始前，图书管理员必须在系统登录成功；

4 后置条件：如果用例执行成功，该读者的借书记录被更新，否则，系统状态不变。



例2：处理销售

1. 顾客携带所购商品或服务到收银台通过POS机付款。
2. 收银员开始一次新的销售交易。
3. 收银员输入商品条码。
4. 系统逐条记录出售的商品，并显示该商品的描述、价格和累计金额。价格通过一组价格规则来计算。
5. 收银员重复3-4步，直到输入结束。
6. 收银员告知顾客总额，并请顾客付款。
7. 顾客付款，系统显示找零信息，并处理支付。
8. 系统记录完整的销售信息，并将销售和支付信息发送到外部的账务系统（进行账务处理和提成）和库存系统（更新库存）。
9. 系统打印票据。
10. 顾客携带商品和票据离开。

用例名称：处理销售

参与者与关注点：

收银员：希望准确、快速地输入，而且没有支付错误，因为如果少收货款，将从其工资中扣除
... ..

前置条件：收银员必须经过确认和认证

成功保证（或后置条件）：存储销售信息。准确计算税金。更新账务和库存信息。

主成功场景（或基本流程）：

1. 顾客携带所购商品或服务到收银台通过POS机付款。
2. 收银员开始一次新的销售交易。
3. 收银员输入商品条码

扩展（或替代流程）：

3a. 无效商品ID（在系统中未发现）：系统提示错误并拒绝输入该ID。
收银员响应该错误

特殊需求：

使用大尺寸平面显示器触摸屏，文本信息可见距离为1米
.....

发生频率：可能会不断地发生

未解决问题：

提成处理规则不确定
收银员换班时如何处理
... ..

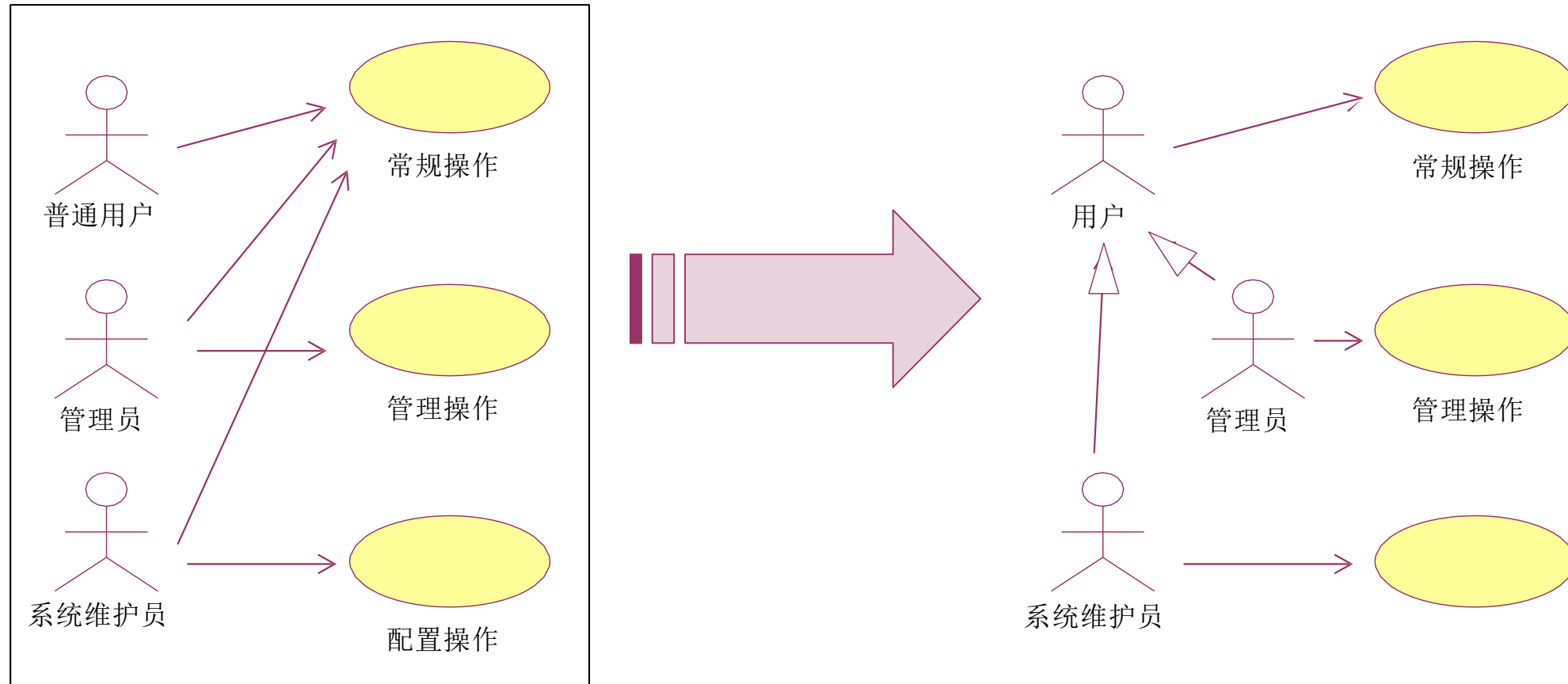


细化用例模型

1. 在一般的用例图中，只需表述参与者和用例之间的通讯关联，除此之外，还可以描述：
 - 参与者与参与者，用例与用例之间的泛化(**generalization**)
 - 用例和用例之间的包含(**include**)
 - 用例和用例之间的扩展(**extend**)
 - 利用这些关系来调整已有的用例模型，把一些公共的信息抽取出来复用，使得用例模型更易于维护。
2. 根据用例描述绘制活动图
3. 补充非功能性需求

泛化(Generalization)

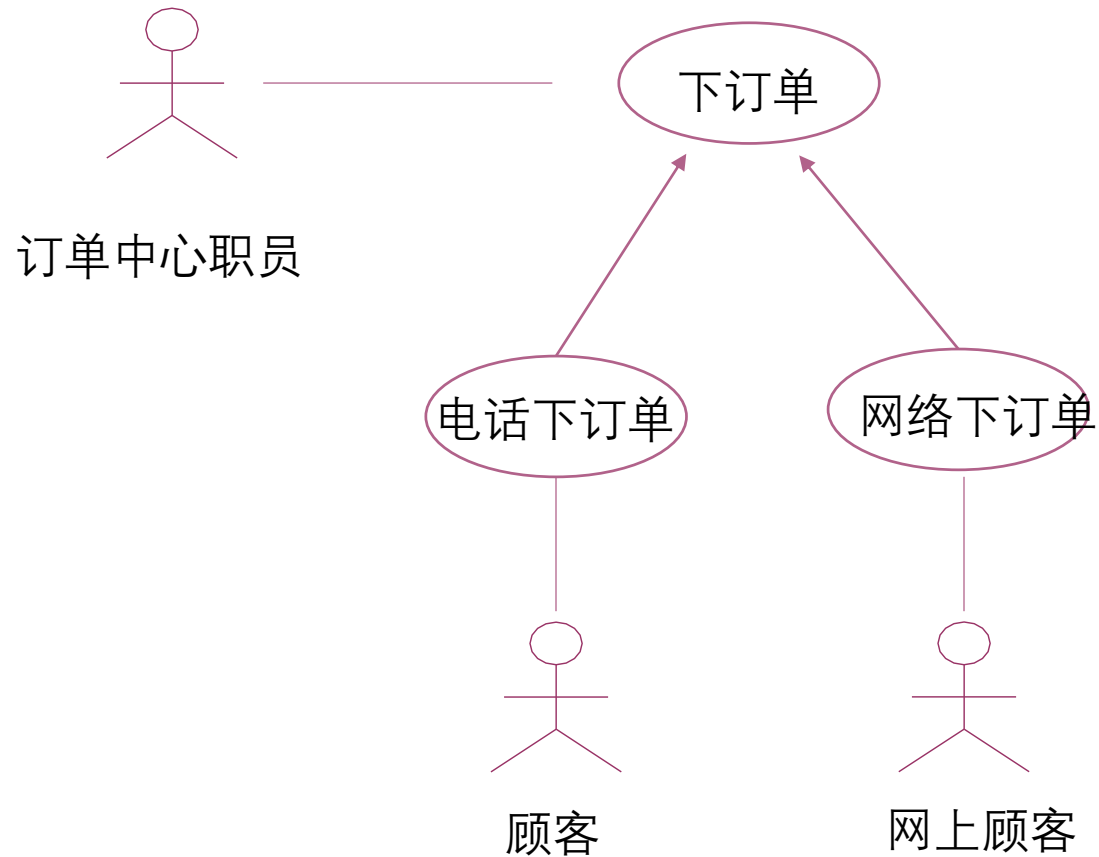
- 参与者之间可以有泛化(Generalization)关系。



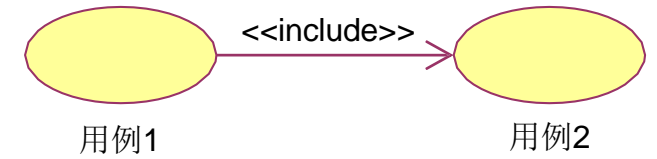


泛化(Generalization)

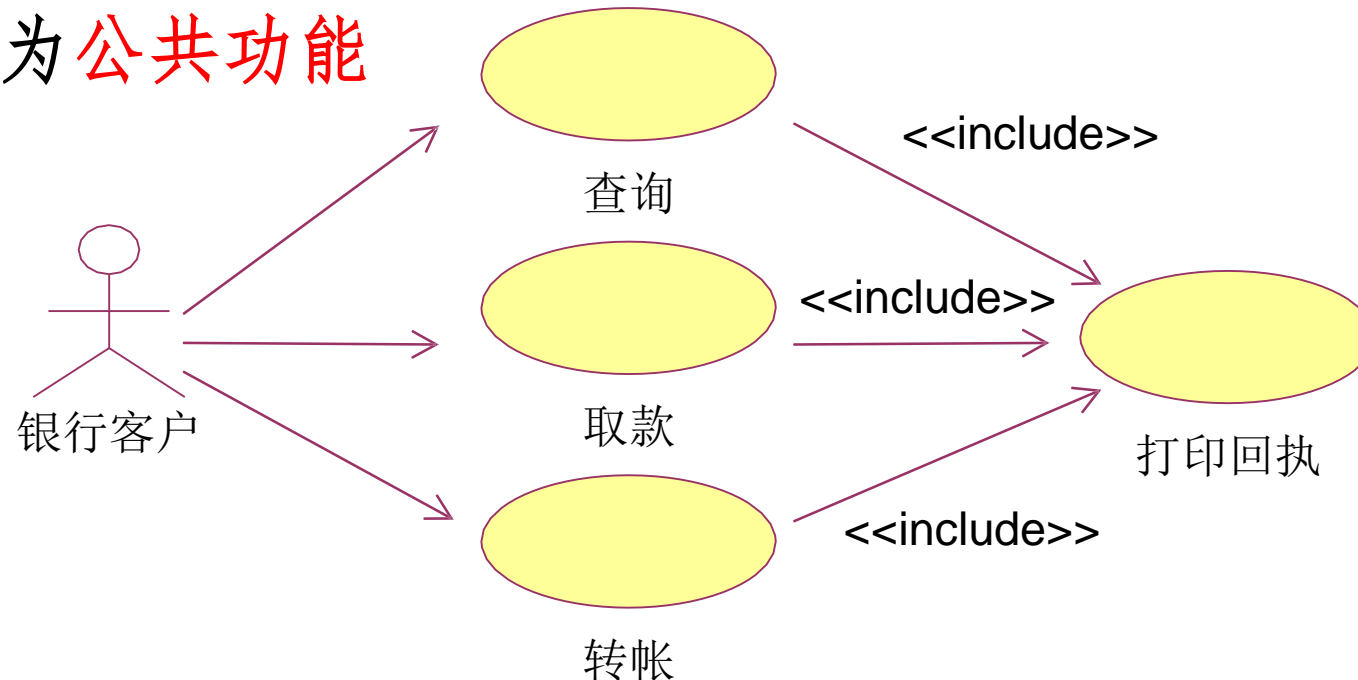
- Case之间也有泛化(Generalization)关系。



包含 (Include)

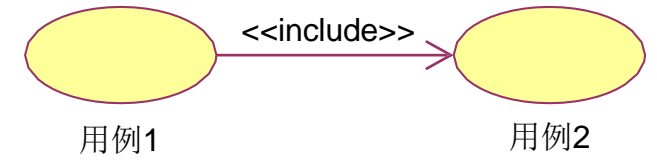


- “包含关系”是通过在关联关系上加入<<include>>标记来表示;
- 语义: 用例1会用到用例2 (无条件执行), 用例2的事件流将被插入到用例1的事件流中
- 一般表示为公共功能

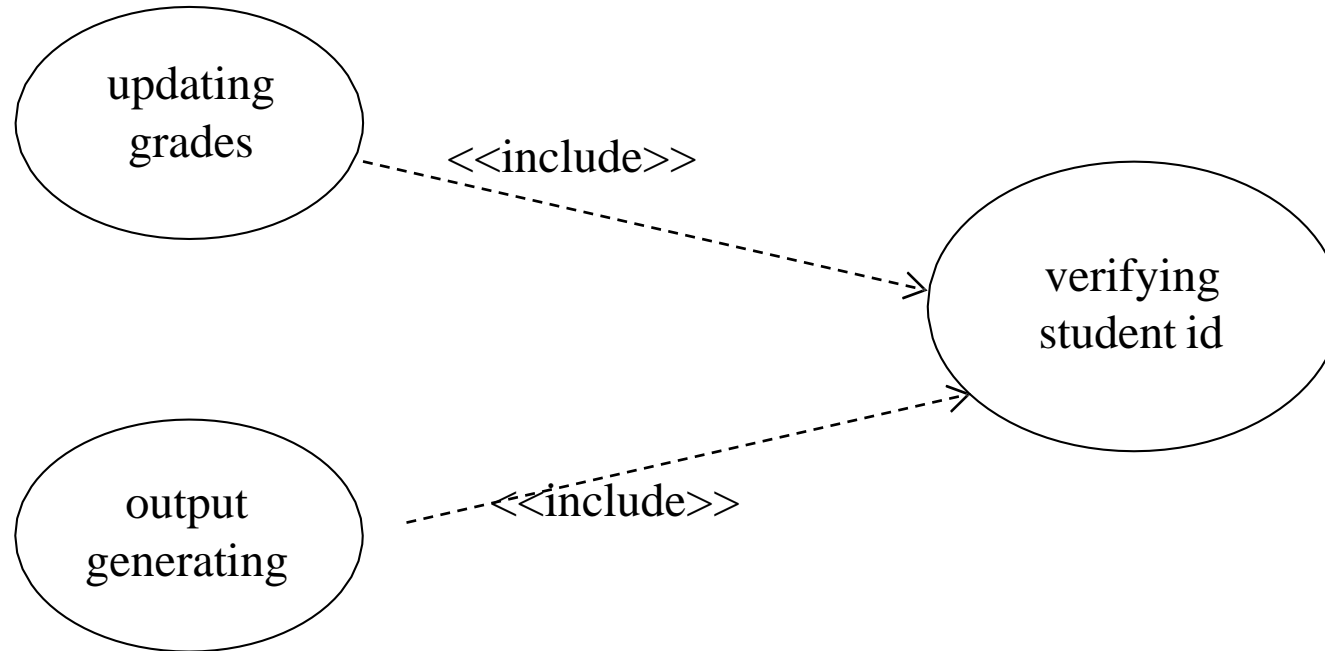




包含 (Include)

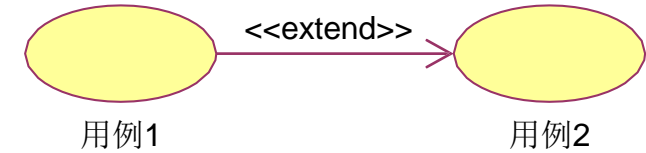


- 通过将公共行为放在用例中，我们可以避免多次描述相同的事件流。

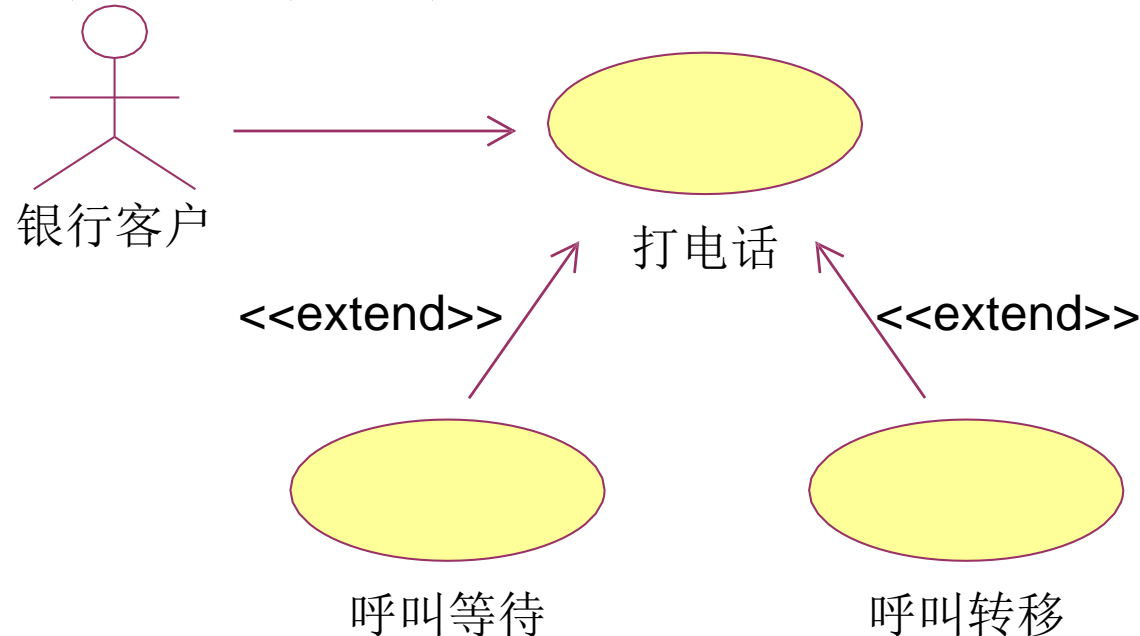




扩展(extend)



- “扩展关系”是通过在关联关系上加入<<extend>>标记来表示；
- 语义：用例2在某些特定情况下（**有条件执行**）会用到用例1，此时，用例1的事件流将被插入到用例2的事件流中。
- 一般表示为**异常功能，大多是扩展流程**





图书管理系统

建立图书信息管理系统，系统要求实现以下功能：

- ✓ 用户管理功能，包括读者信息的录入、修改、更新以及登录
- ✓ 书籍管理功能，如书籍的添加、修改、更新、删除等数据维护功能，还可以根据读者借阅书籍的要求随时更新图书馆的书籍数据库
- ✓ 书籍的借阅、归还管理、如借还进行详细登记，更新书籍数据库。同时提供图书预定功能
- ✓ 信息查询功能，如图书信息查询、用户借书、还书信息查询、书籍库存情况查询等

请根据以上描述，确定执行者及用例，建立系统的用例模型。

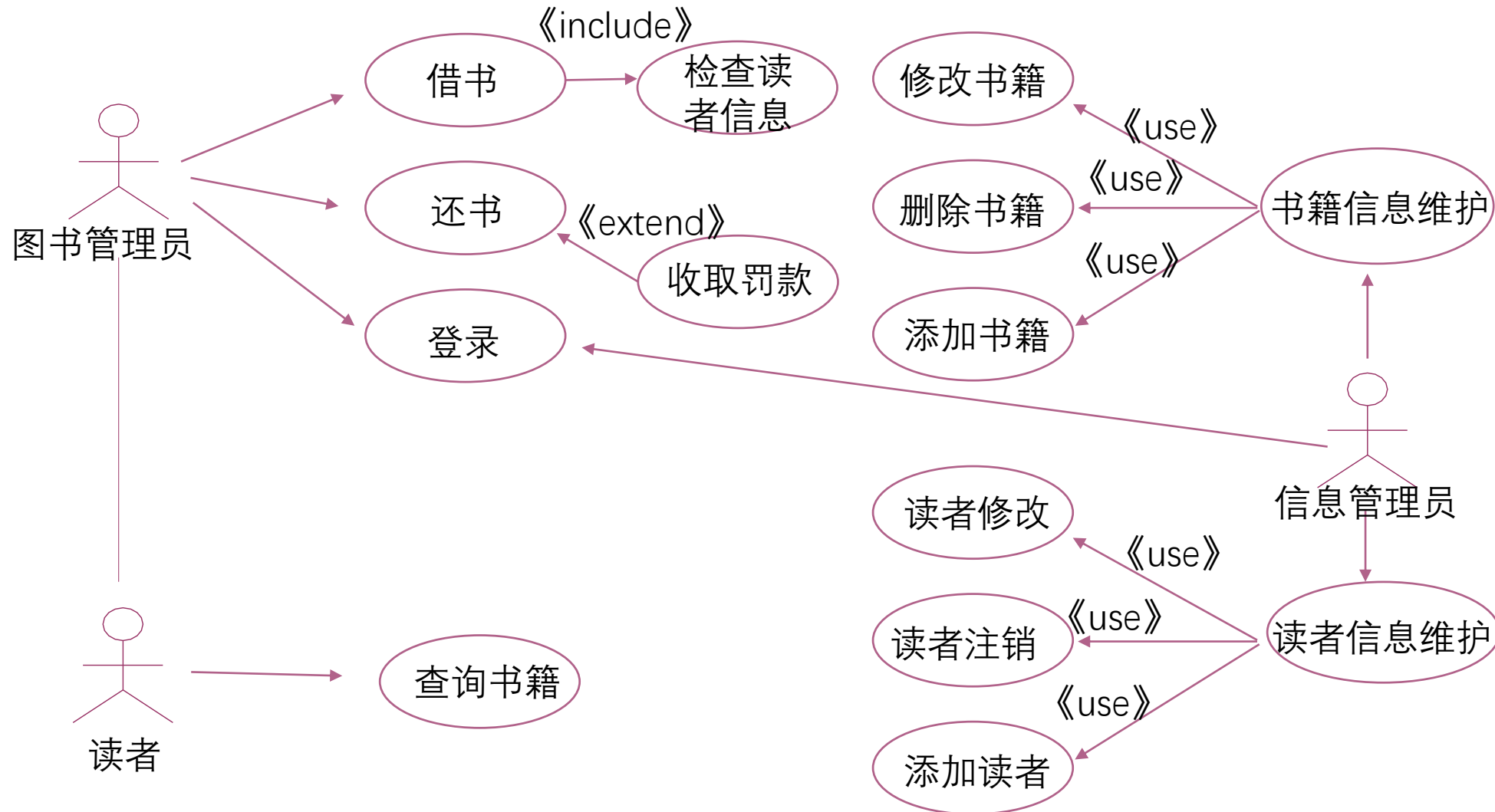


图书管理系统

- 参与者：管理员和读者；管理员=后台信息维护管理员+图书管理员
- 读者可以查询书籍获得书籍信息。
- 读者可以通过管理员来进行图书的借、还操作
- 图书管理员可以执行借书和还书操作，需要用管理员账号进行登录。
- 信息维护管理员可以进行书籍信息维护和读者信息维护，需要用管理员账号进行登录。
- 书籍信息维护主要包括添加书籍、修改书籍和删除书籍
- 读者信息维护主要包括添加新读者、读者信息修改和读者注销



图书管理系统用例图





本章內容

- 用戶故事 (User Story) 案例
- 用例圖 (User Case Diagram)
- 活動圖 (Activity Diagram)



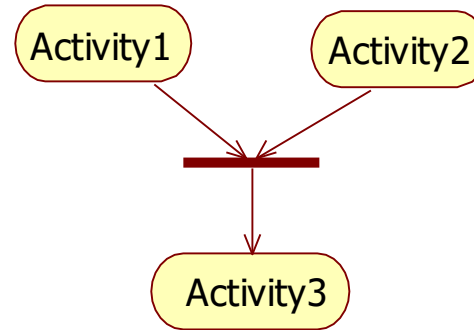
活动图

■ 什么是活动图？

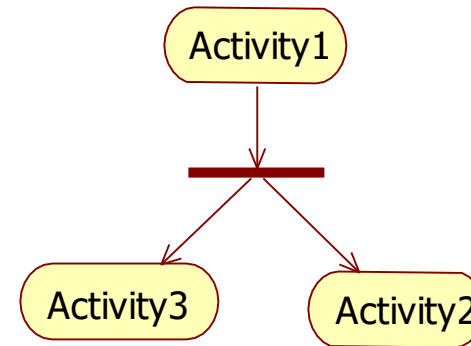
- 显示了组成复杂过程的步骤序列，例如算法或 workflows
- 活动图用于详细描述用例
- 是状态图的一个变种
- 活动图的目的是描述动作及动作的结果
- 活动图中的动作可以放在“泳道中”，泳道聚合一组活动，通常根据活动的功能来组合活动。



活动图图形元素



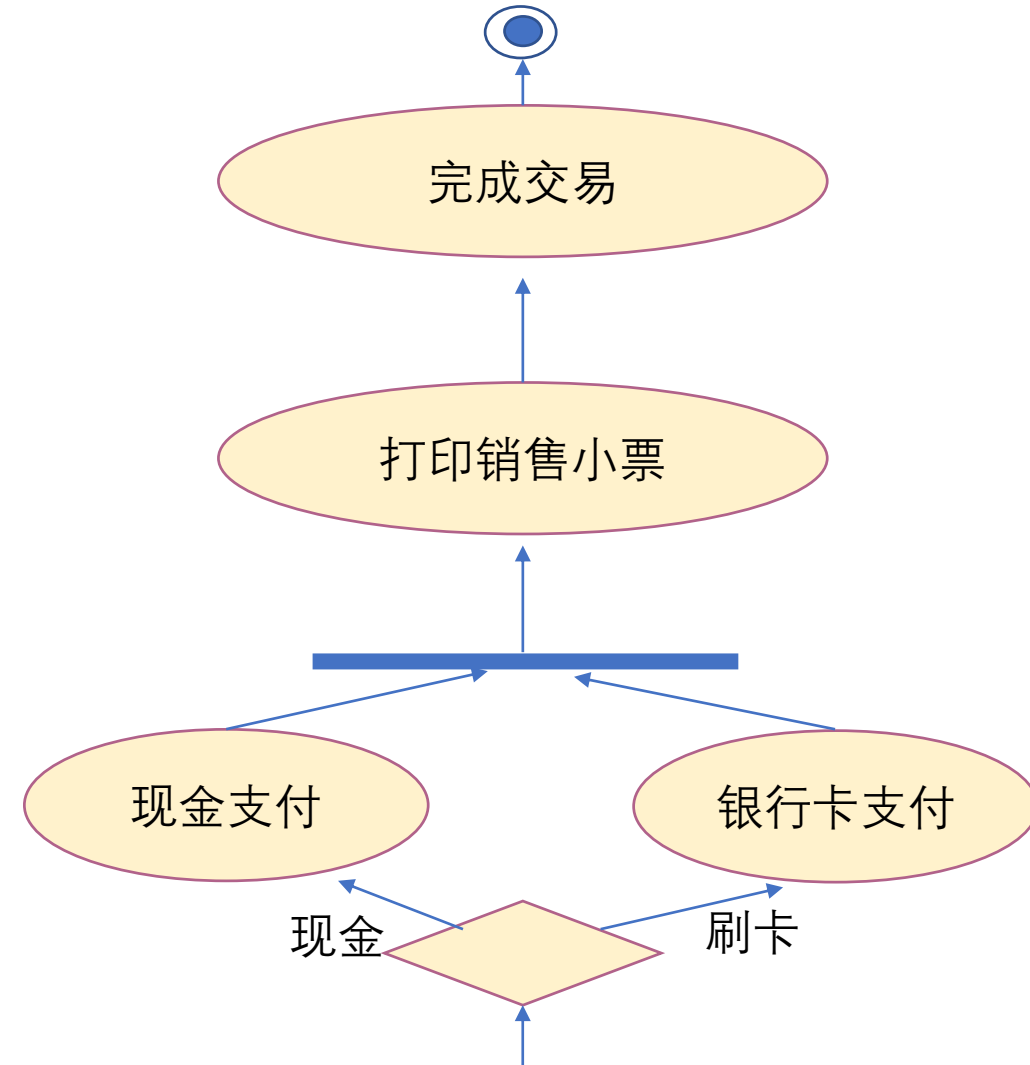
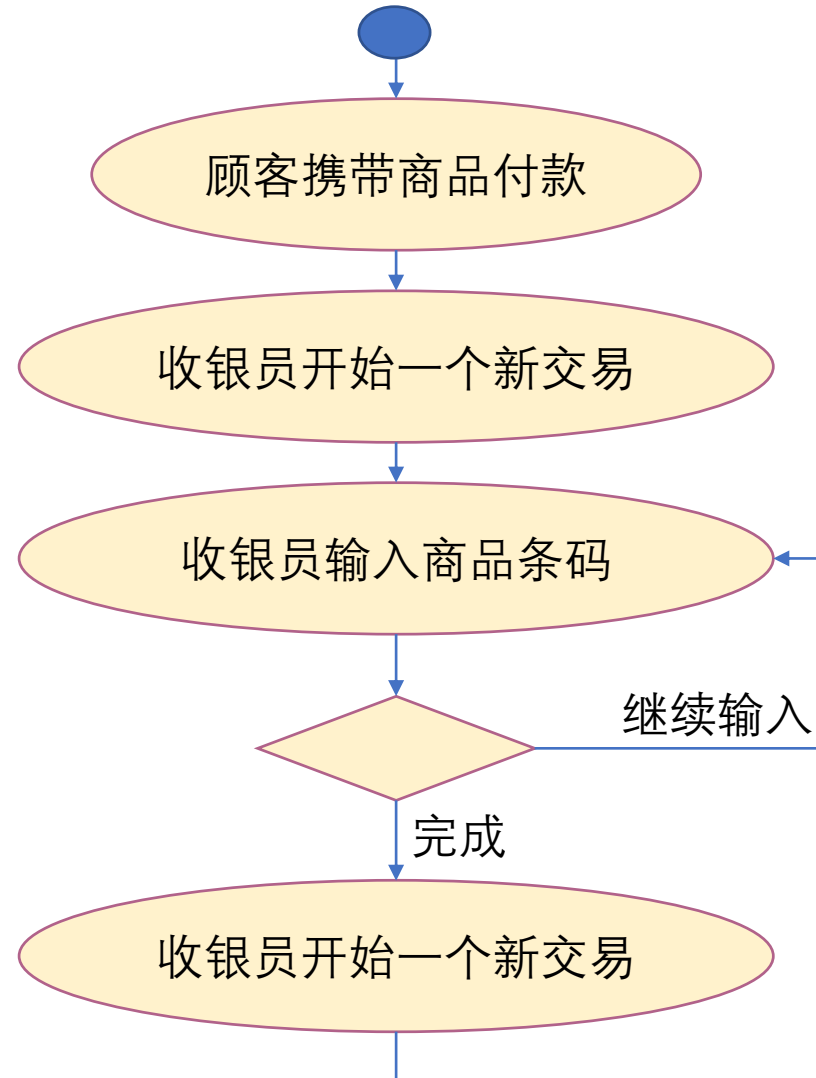
并发合并



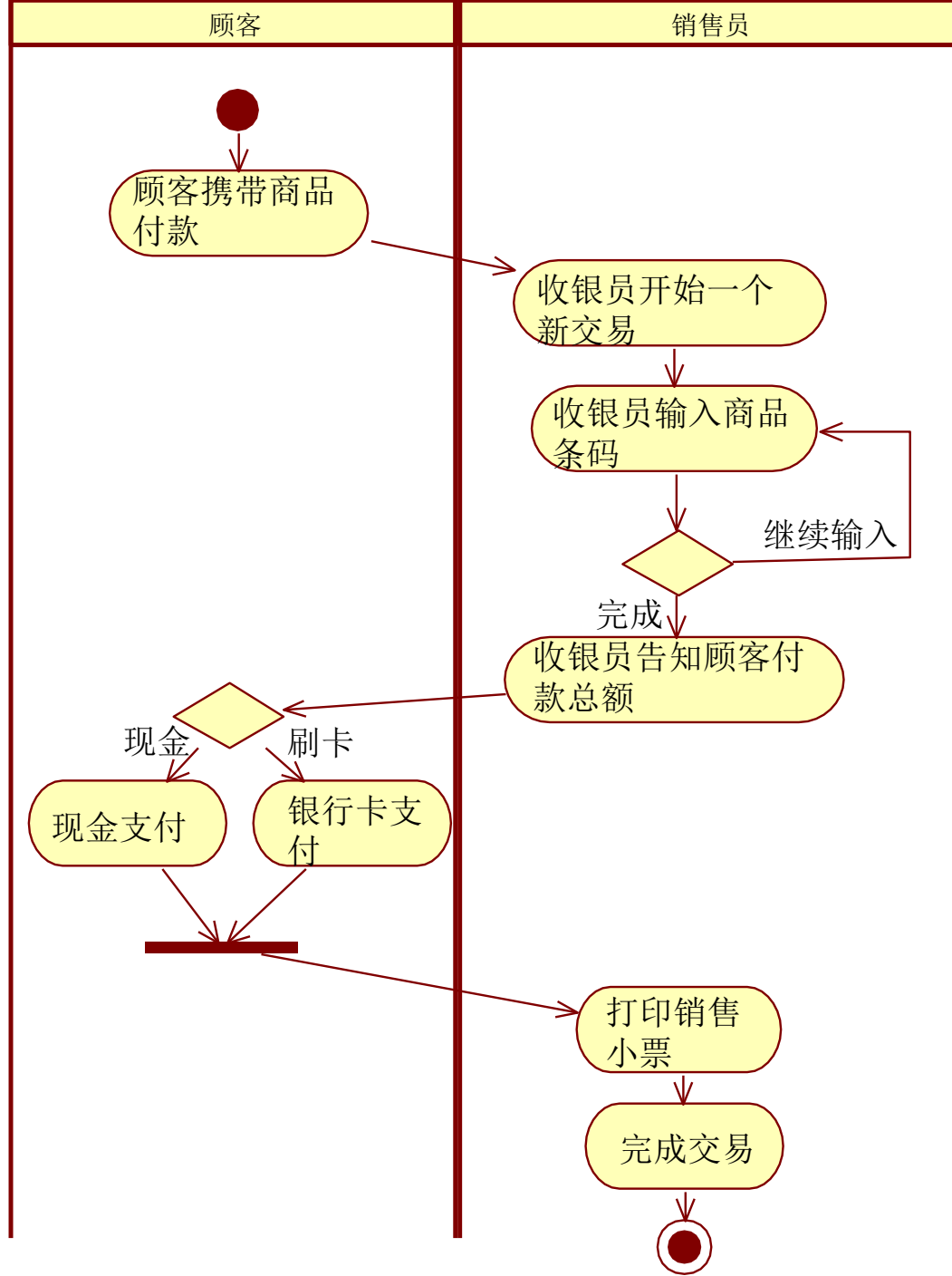
并发分叉



活动图描述

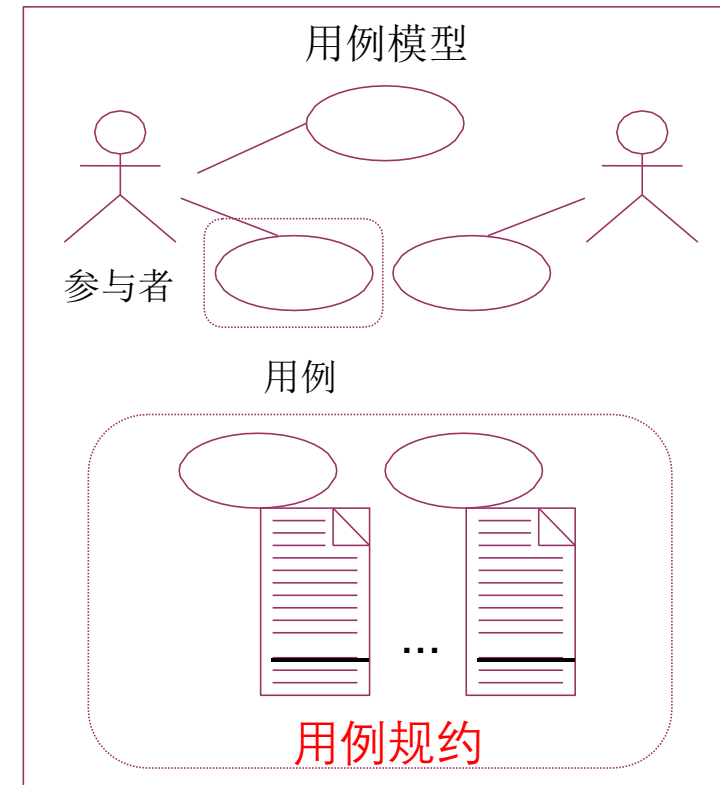


泳道图



最终用例模型的提交物

- 1 用例模型
- 2 每个用例的详细描述
- 3 术语表：所用到的术语说明
- 4 补充规约：非功能性需求的说明





课后作业

- 以摩拜单车系统为例，尽自己最大可能画出系统用例图，并以用户租用单车和管理员增加单车为例做详细的用例描述