

需求工程

第三讲

鲁赞娜·奇钦，乔恩·伯德，皮特·班尼特

助教: 亚历克斯·埃尔伍德，亚历克斯·科克里安，卡斯帕·王

概观

- 什么是需求？
- 利益相关者识别
- 功能性和非功能性需求
- 用用例模型描述系统行为并在模型中捕获它
 - 用例图
 - 用例规格
- 需求质量

要求

- 系统需求指定一个系统，不是从系统实现的角度，而是从用户观察的角度。需求记录对系统功能和约束的描述。
 - 他们说，功能需求规定了用户与系统的交互系统应该做什么：
 - 系统应提供的服务声明
 - 系统应该如何对特定输入做出反应
 - 系统在特定情况下应该如何运行
 - 也可以说明系统不应该做什么
 - 非功能性需求指定了其他系统属性，它们说明了功能性需求是如何实现的：
 - 对系统提供的服务或功能的限制
 - 通常适用于整个系统，而不仅仅是单个功能

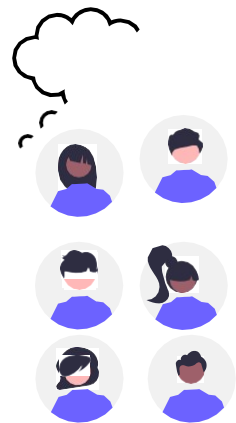
为什么我们需要需求工程？

General Problems and Requirements

- 术语不一致:人们用自己的话表达需求
- **对同一系统的冲突需求**
- 人们经常不知道自己需要什么（或者至少不知道解释一下！）
- 需求经常变化
- 相关人员/信息可能无法访问

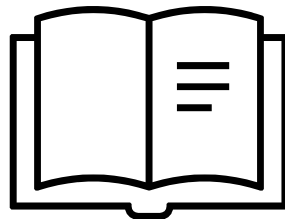
Requirements are communication

系统管理员



客户

建议



要求

写

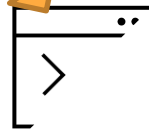
商业分析员

阅读
固定



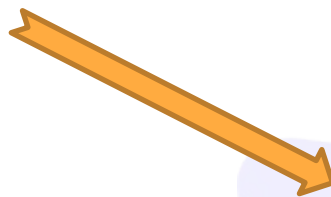
开发商

写



软件

将软件与需求
进行比较



测试员

Requirements are

- 独自或两人一组
- 按照一些说明进行绘制。
- 然后我会判断你是否正确地遵循了指示。
- 不是你的艺术技巧！

Drawing Activity: 5

Requirements are **acceptance**

- 为了能够公平地评估团队是否产生了匹配的东西
你要的东西，你要的东西必须是：
 - 明确/精确
 - 完成
 - 可以理解/清楚

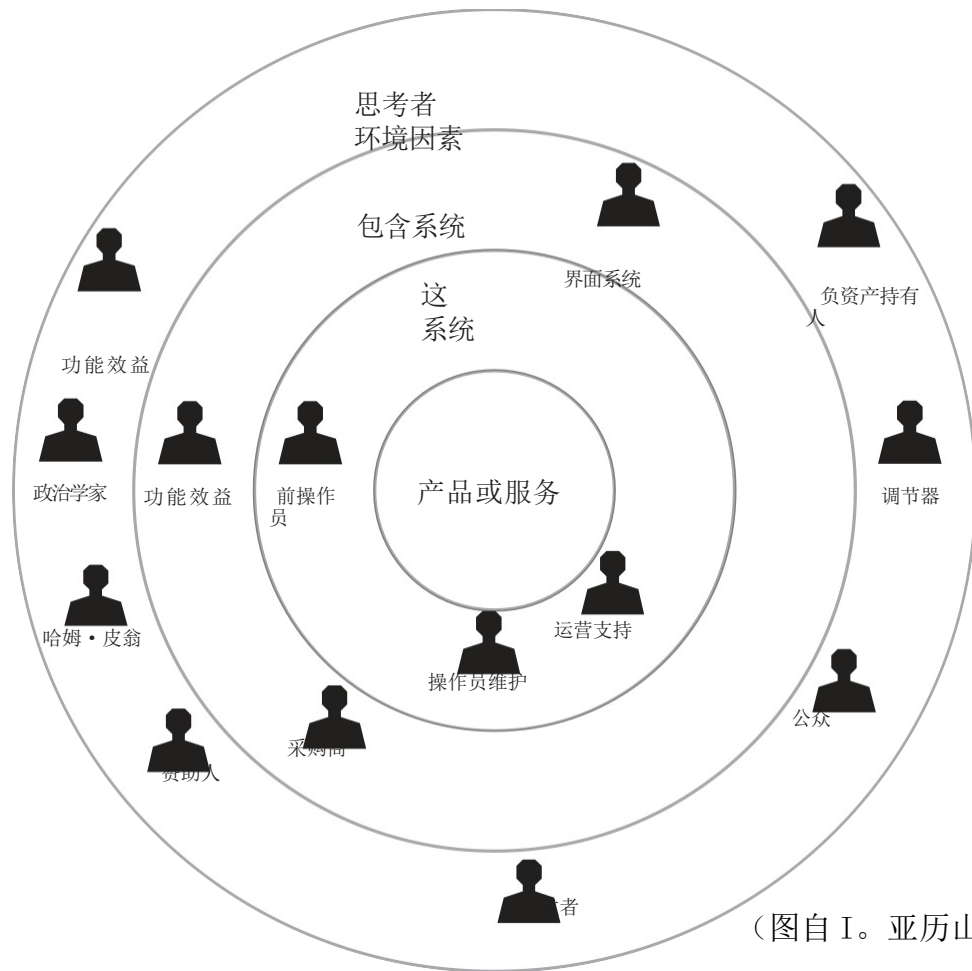
Analysing

1. 确定与系统相关的利益相关者
2. 确定顶级用户需求（如 nfr 或“用户故事”）
3. 将故事分解成单独的步骤/细化需求
4. 指定原子需求（例如，用户故事中的每一步）让我们依次看看每

个阶段…

1. 确定利益相关者

洋葱模型



(图自 I. 亚历山大的书)

利益相关者

识别

- 客户
- 文件，如组织结构图
- 模板（例如洋葱模型）
- 类似项目
- 分析项目背景

请记住：

- 代理利益相关者（例如，法律、目前不可用、大规模产品用户）
- 负面利益相关者

2. 确定最高级别的需求/问题

确定“用户故事”

记录用户需求的流行方式…

作为《用户类型》，我想要《某个目标》
所以《某种原因》。

作为一名学生，我希望能够注册一个模块，这样我就可以了解我感兴趣的主题。

作为客户，我希望能够支付大学课程的费用，这样我就可以参加讲座以获得学位。

作为客户，我希望安全地保存我的数据，以保护我的隐私。

What Is System

- 系统行为是系统如何行动和反应。
 - 它由系统的动作和活动组成。
- 系统行为在用例中被捕获。
 - 用例描述了系统和（部分）环境之间的交互。

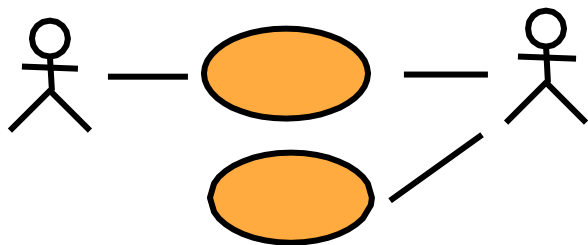
What is a use-case

- 根据用例描述系统的功能需求
- 将涉众需求与软件需求联系起来
- 充当规划工具
- 由参与者和用例组成

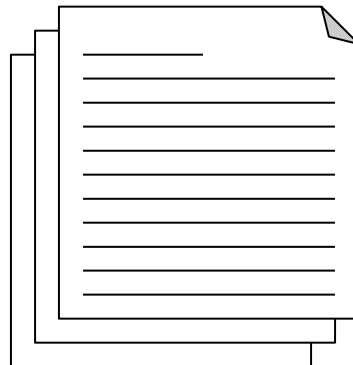
Capture a use-case

- 用例模型包括：

用例图
(视觉表现)

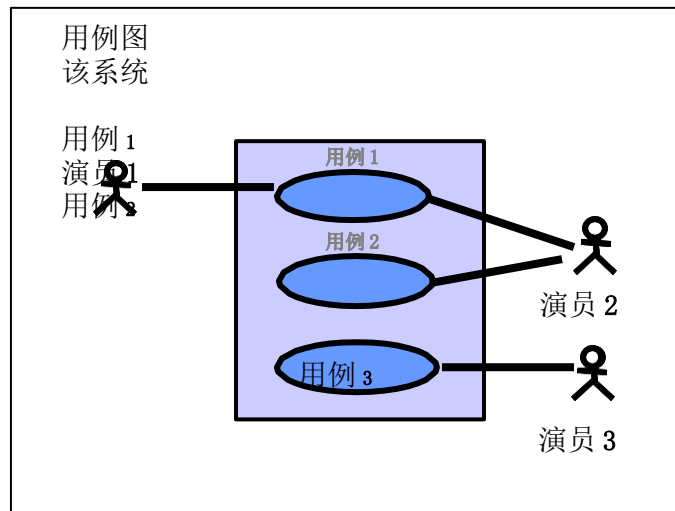


使用案例
规范
(文本表示)

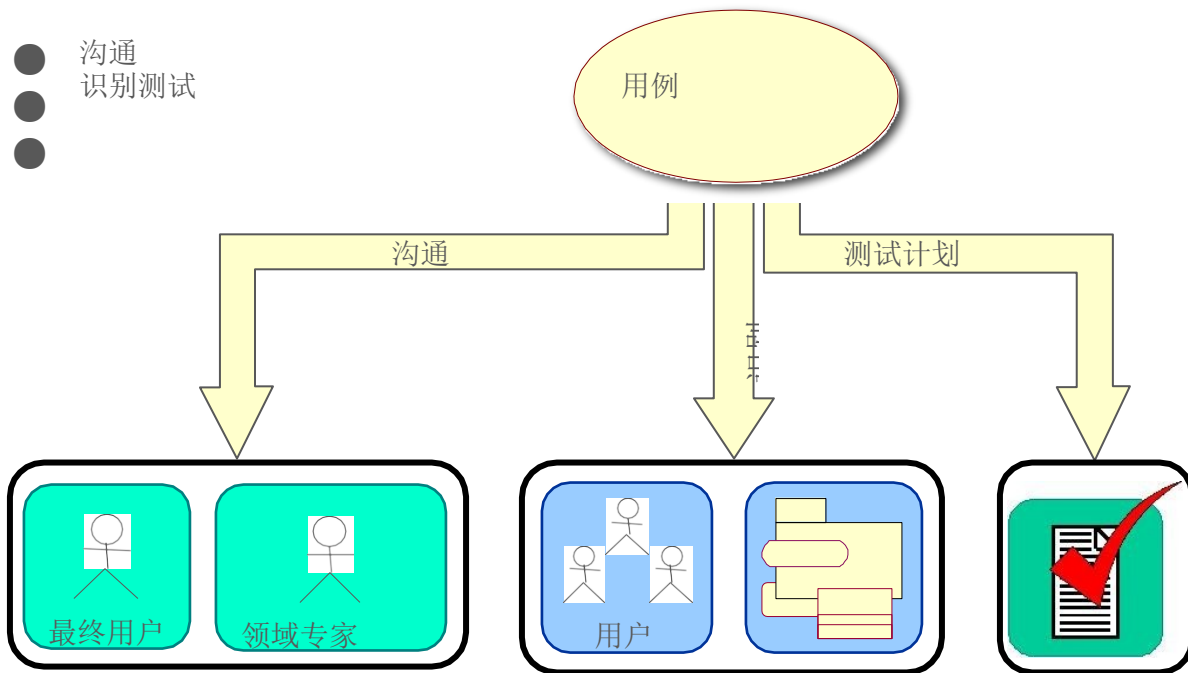


Use-case

- 展示了一组用例、参与者及其关系
- 定义系统的清晰边界
- 确定谁或什么与系统交互
- 总结了系统的行为



What Are the Benefits of a Use-Case

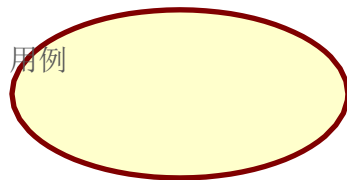


Major Concepts in Use-Case

- 参与者代表任何与系统交互的事物。
- 一个用例描述了由系统执行的一系列事件，这些事件产生了对特定参与者有价值的可观察结果。



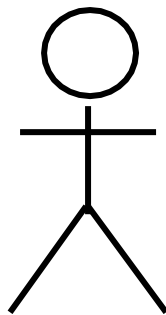
演员



用例

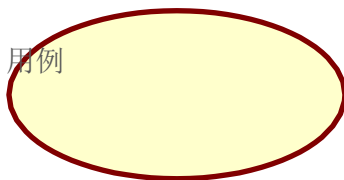
What Is an

- 参与者代表系统用户可以扮演的角色。
- 他们可以代表一个人、一台机器或另一个人系统。
- 他们可以主动与系统交换信息。
- 他们可以提供信息。
- 他们可能是信息的被动接受者。
- 演员不是系统的一部分。
 - 演员是外在的。演员



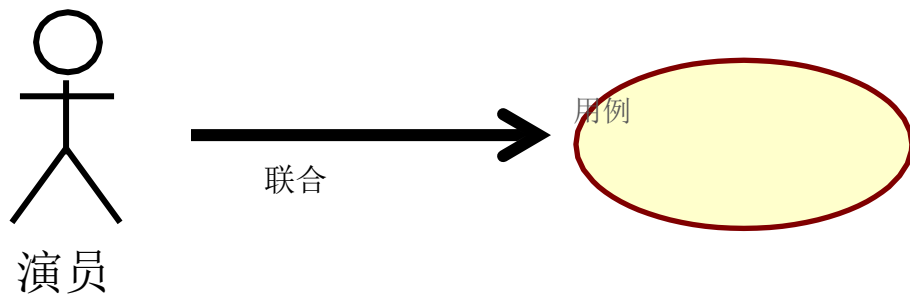
What Is a Use

- 定义一组用例实例，其中每个实例都是系统执行的一系列操作，这些操作为特定参与者产生了可观察到的价值结果。
 - 用例为一个或多个参与者之间的对话建模系统呢
 - 用例描述了系统采取的行动向演员传递有价值的东西



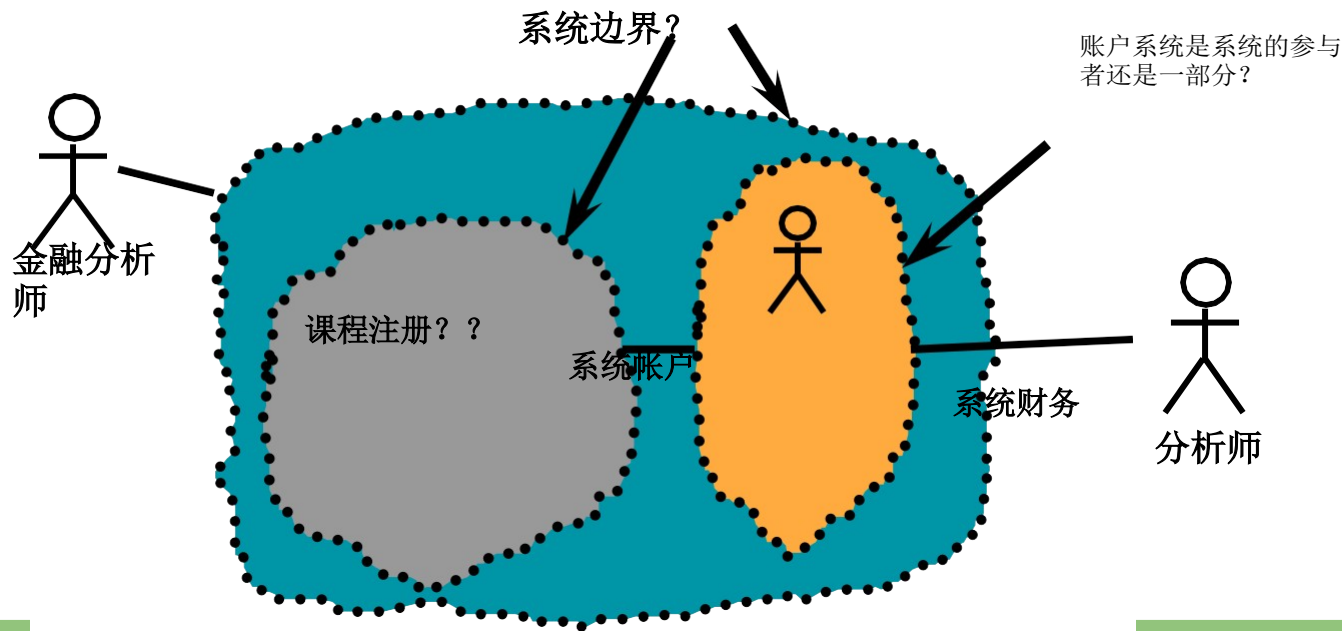
用例与参与者

- 用例为参与者和系统之间的对话建模。
- 一个用例由一个参与者发起，以调用系统中的某个功能。



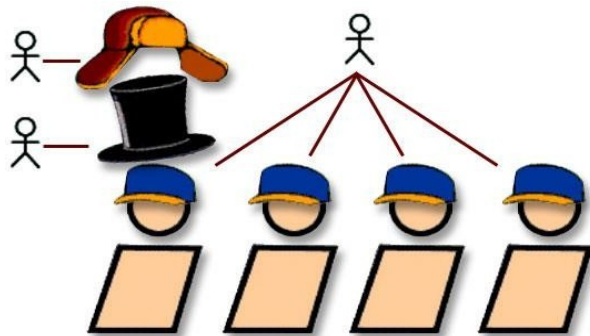
行动者和系统边界

- 确定系统边界是什么
- 边界之外与系统交互的所有内容都是演员

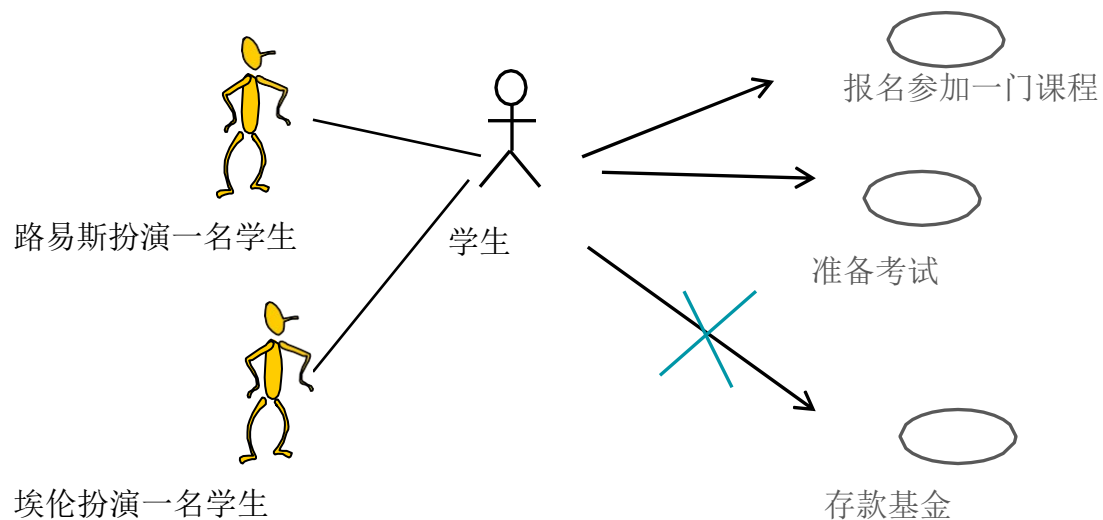


演员和角色

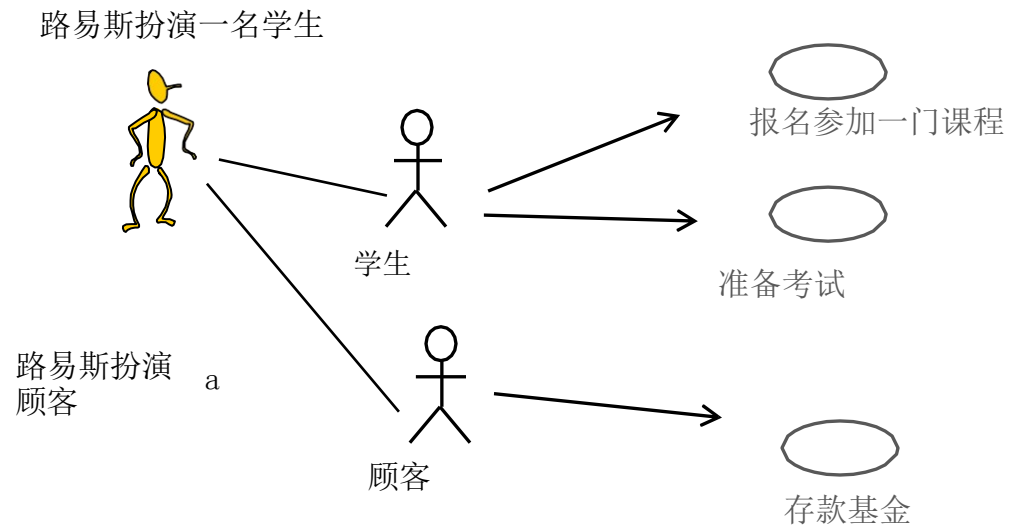
- 参与者代表一个人、硬件设备或另一个系统可以根据系统进行规划。



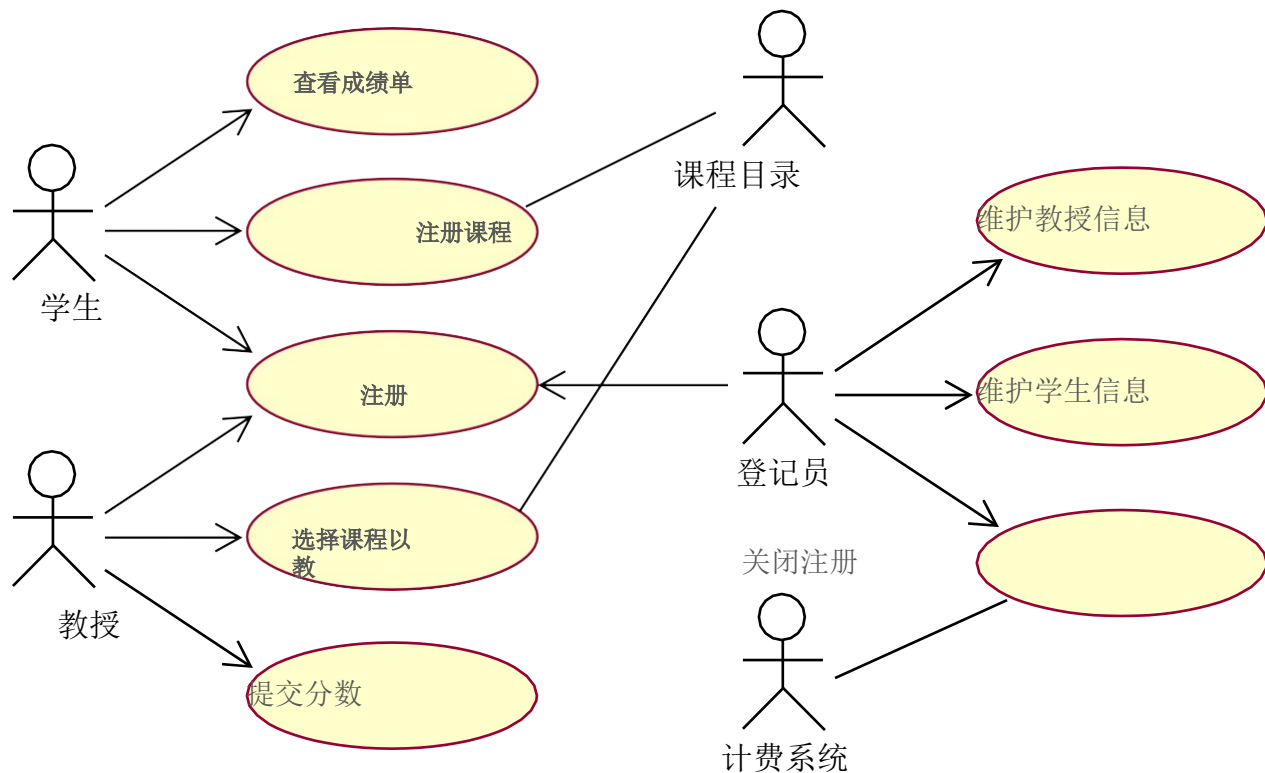
Actor



Actor

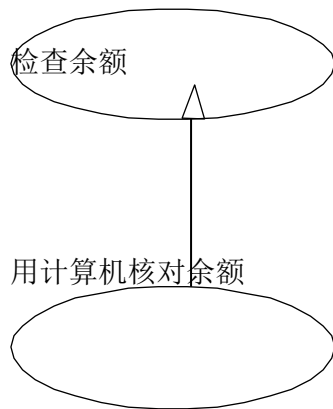


你如何阅读这张图表？

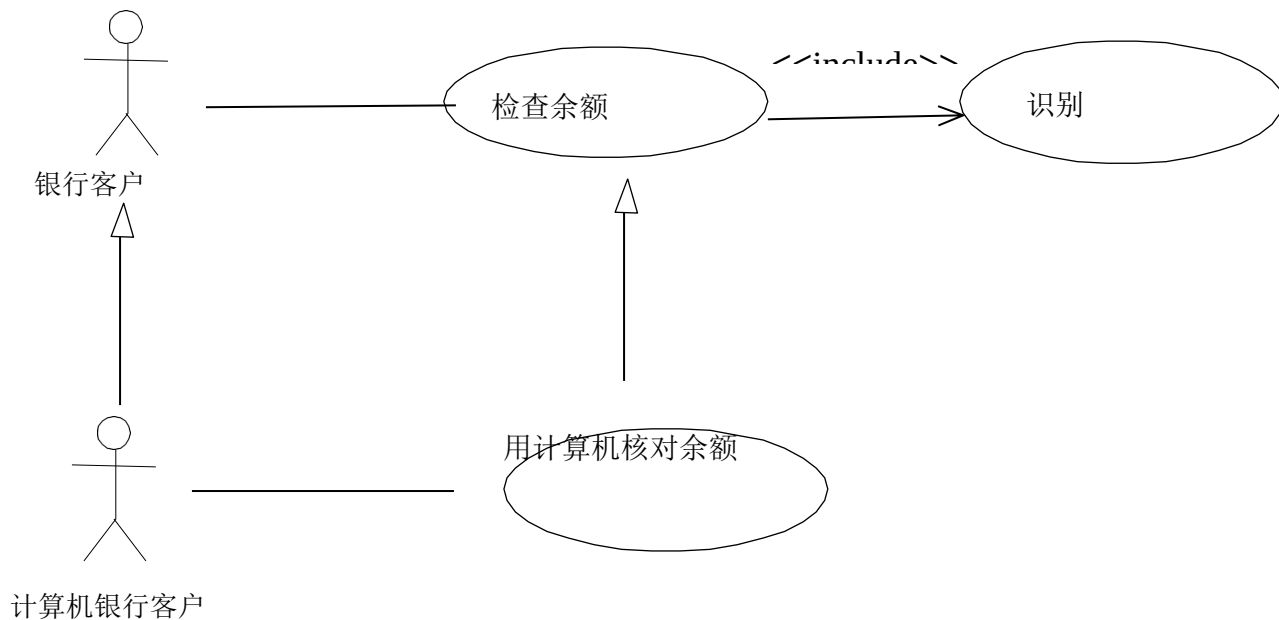


Modelling with Use

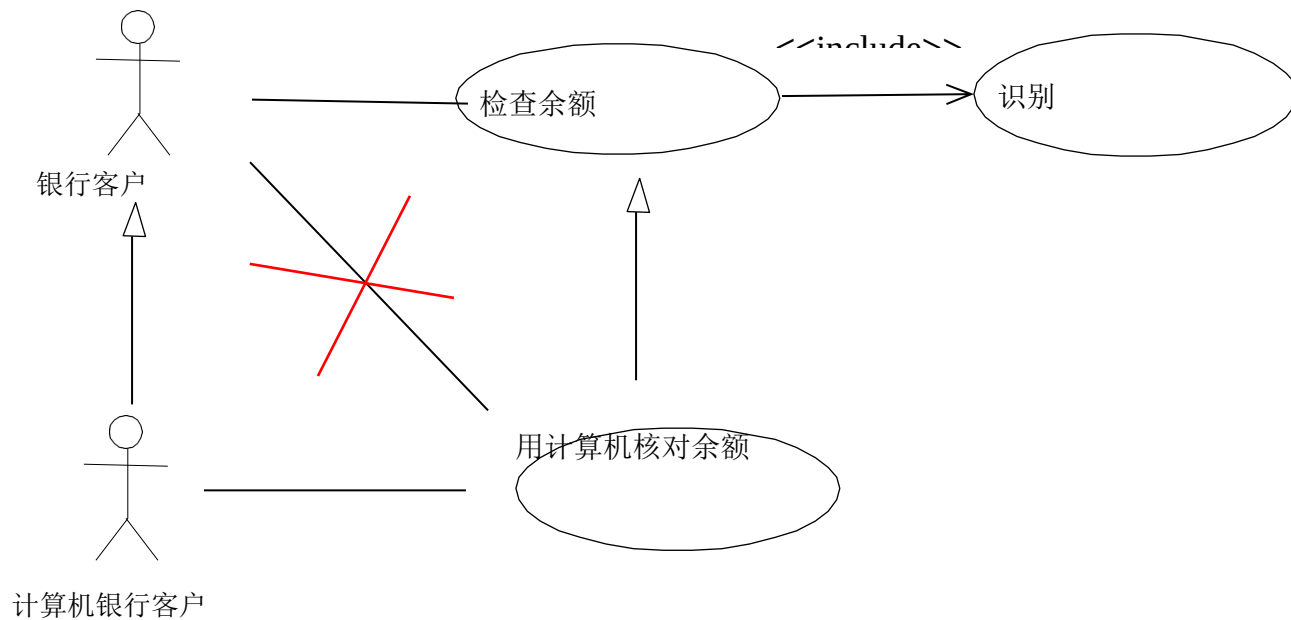
- 用例之间的泛化意味着子用例比父用例更具体；子对象继承父对象的所有属性和关联，但可以添加新功能



Structuring Use Case Diagrams

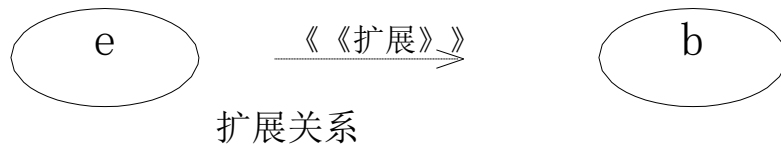


Use Case Diagram: Structuring

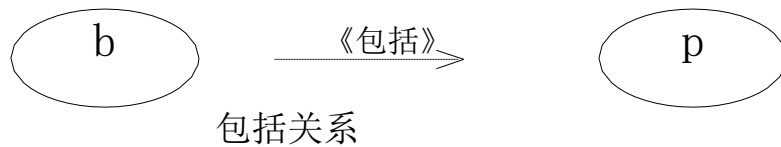


Use

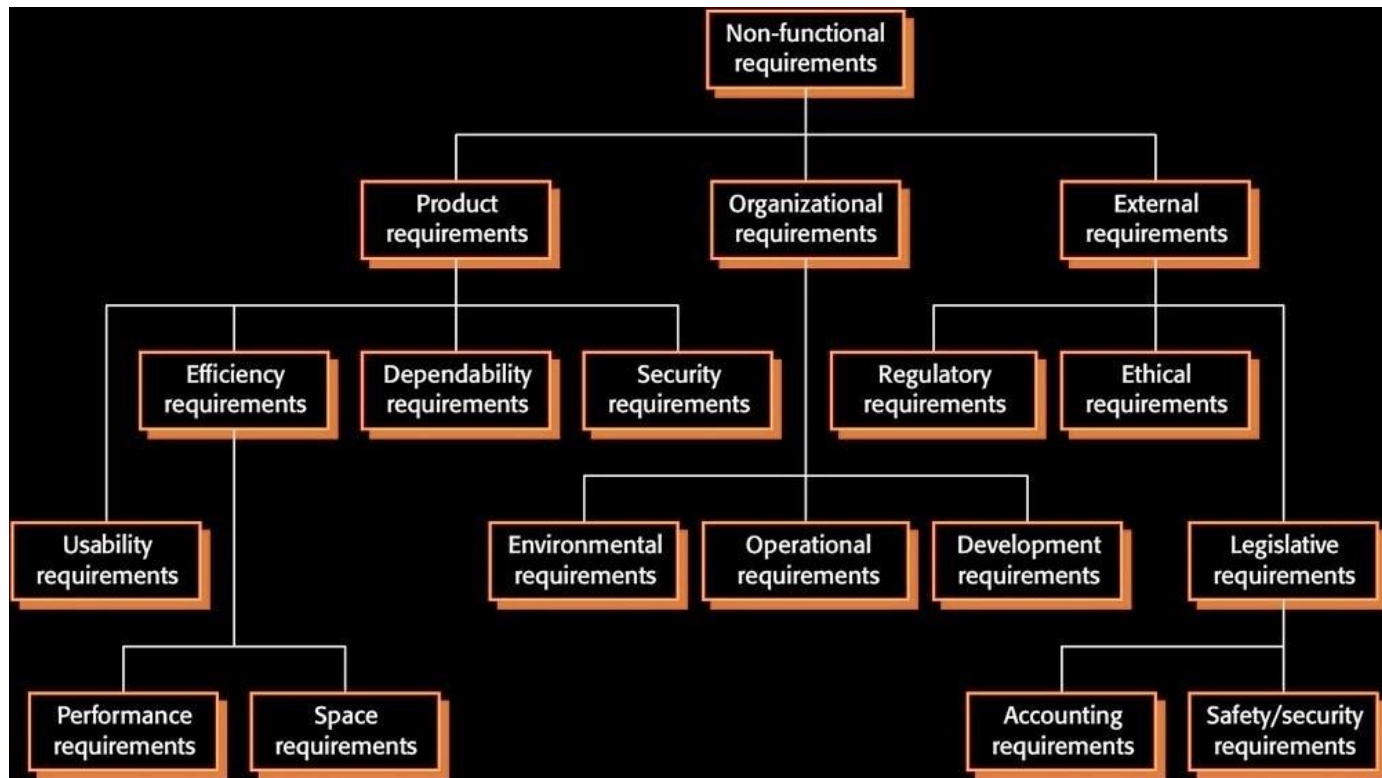
指定如何将扩展用例 e 的行为插入到基本用例 b 的行为中。
e 是可选的。



指定包含的用例 p 的行为如何影响基本用例 b 的行为。



还有:非功能性需求



3. 将故事分解成单独的步骤/细化需求

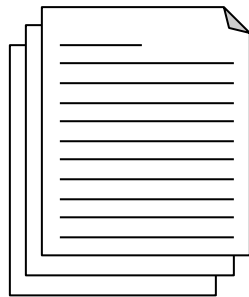
用例规格说明

- 包含以下内容的需求文档

用例的文本，包括：

- 描述参与者和系统之间交互的事件流的描述
- 其他信息，如：
 - 前提
 - 后置条件
 - 特殊要求
 - 关键场景
 - 子流程

用例规格说明



概述每个使用案例

- 大纲用简短的句子捕获用例步骤，并按顺序组织

给步骤编号并命名



用例名称简要描述基本流程

第一步

第二步

第三步

替代流程

备选流程 1

备选流程 2

备选流程 3



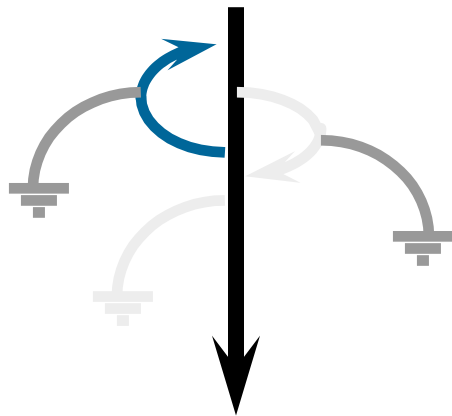
将基本流程组织成步骤



确定备选流程

事件流（基本和备选）

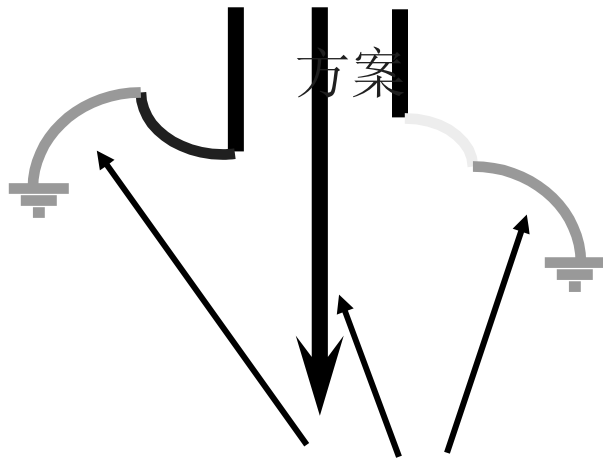
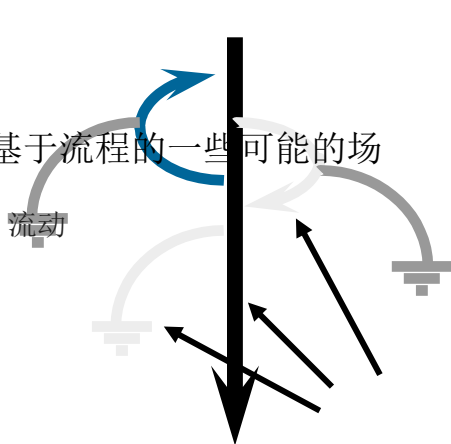
- 流程是一系列步骤
- 一个基本流程
 - 从头到尾的成功场景
- 许多可选流程
 - 常规变体
 - 奇怪的案件
 - 异常（错误）流程



什么是用例场景？

- 一个用例的实例
- 从开始的一组有序操作
用例到它的一个端点

注：此图仅说明了基于流程的一些可能的场景。



用例的检查点

- ✓ 每个用例都是独立于其他用例的
- ✓ 没有用例有非常相似的行为或事件流
- ✓ 事件流的任何部分都没有被建模为另一个用例

4. 指定原子需求（例如，用户故事中的每一步）

- ✓ 本课程中未详细介绍，例如：
 - ✓ 结构化语言
 - ✓ 形式方法

Quality Attributes of

- 一致性:需求之间有冲突吗?
- 完整性:所有特性都包含在内了吗?
- 可理解性:需求能被理解吗?
- 可追溯性:需求的来源被清楚地记录了吗?
- 现实主义:给定可用的资源和技术,可以实现需求吗?
- 可验证性:需求可以被“勾掉”吗?

Verifiability of

我们应该确保需求是可验证的
没有必要指定一些不能“打勾”的东西

例如，以下是一个无法验证的“目标”：
该系统必须易于等待人员使用，并应组织有序，以最大限度地减少用户错误。

相比之下，以下是可测试的要求：
经过四个小时的培训后，等候人员应能够使用所有系统功能。培训结束后，有经验的用户每小时使用系统的平均错误次数不得超过两次。

Requirements Elicitation

- 面谈
- 观察
- 调查
- 当前文档
- 类似的产品和解决方案
- 合作设计
- 样机研究
- ...

Review

- 模特是干什么的？
- 什么是系统行为？
- 什么是演员？
- 一个用例？
- 什么是角色？
- 我们如何知道我们的需求质量是否良好？

