北京航空航天大學
BEIHANG UNIVERSITY

信息系统分析与设计

用例建模 Use Case

信息系统系　刘冠男

数据智能研究中心
RESEARCH CENTER FOR
DATA INTELLIGENCE

用例建模

- **构建一个软件系统最困难的部分是正确地确定要构建什么**
- **理解关联人员的需求和开发系统的原因：以用户为中心开发**
- **鼓励用户参与**
  - 提供捕捉功能需求的工具
  - 将系统范围分解：分而治之
  - 提供有效交流的工具
  - 辅助估计项目范围、投入和进度
  - 需求跟踪工具
  - 设计用户和系统接口的功能规格说明
  - 提供了定义数据库访问需求的手段
  - 提供了驱动系统开发的一个框架

用例建模主要任务

- 确定系统边界
- 确定参与者
- 确定用例
  - 描述用例
  - 创造场景
- 确定参与者与用例的关系

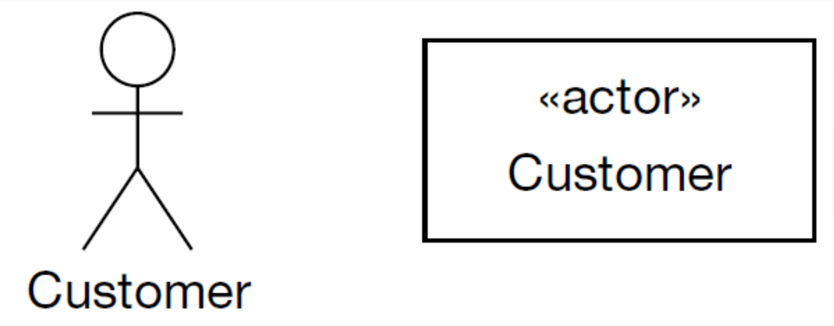
确定系统边界

- 首要任务
- 需要定义:
  - 谁使用系统？
  - 系统为使用者提供何种功能？
- 系统边界的图示:
  - Box labeled with the name of the system
  - Actors outside and use cases inside.
- 系统边界的确定是一个不断细化的过程

确定参与者

- 参与者是直接与用户发生交互的角色.

UNIFIED
MODELING
LANGUAGE™



Usually for people

Usually for other systems

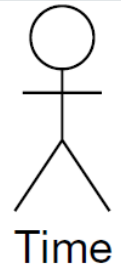
- 参与者位于系统的外部.

确定参与者 cont’d

- **Identifying actors:**
  - Who or what uses the system?
  - What roles do they play in the interaction?
  - Who installs the system?
  - Who starts and shuts down the system?
  - Who maintains the system?
  - What other systems interact with this system?
  - Who gets and provides information to the system?
  - Does anything happen at a fixed time?

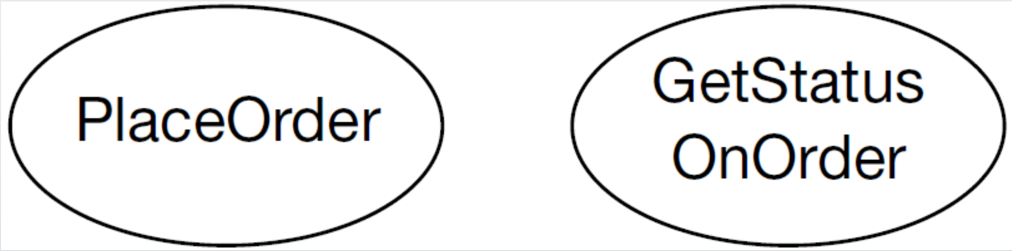
确定参与者cont’d

- **参与者始终位于系统外部**
  - 不受控制的
- **参与者直接与系统交互**
  - 如何帮助定义系统边界
- **参与者是一个泛指，不针对特定的人或物**
- **Time as an actor**
  - e.g., automatic system backup that runs every evening



确定用例

- **用例的定义(*UML Reference Manual*),**
  - “A specification of sequences of actions, including variant sequences and error sequences, that a system, subsystem or class can perform by interacting with outside actors.”
- **参与者希望系统能做的事情**

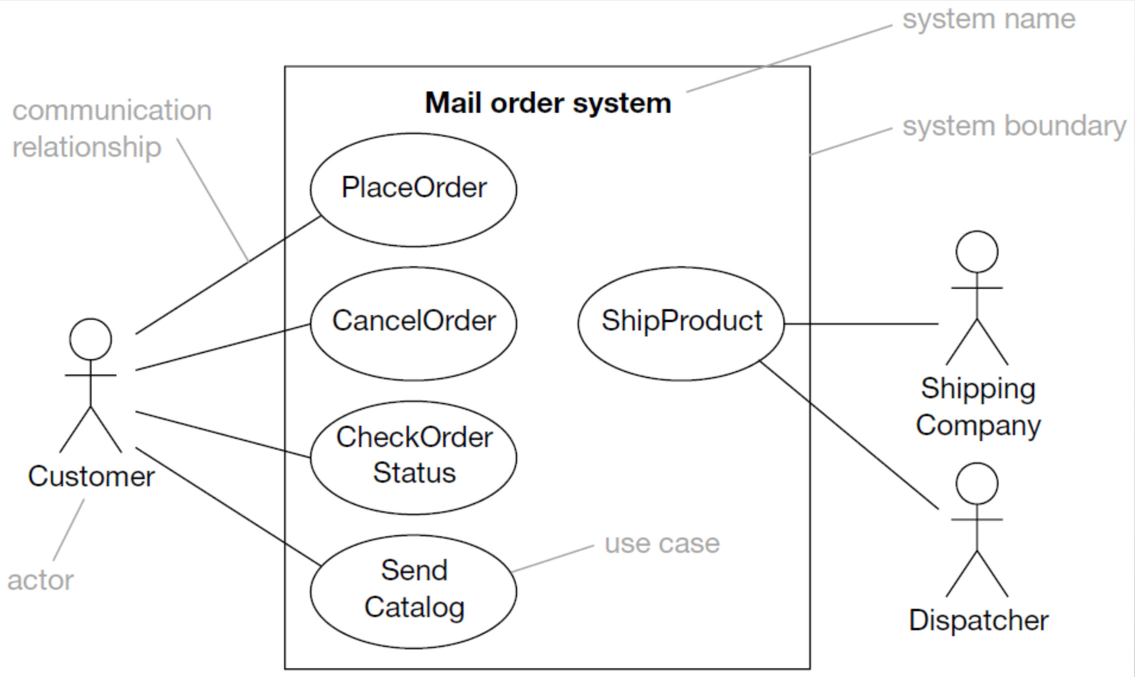


确定用例 cont’d

- **Identify use cases:**
  - What functions will a specific actor want from the system?
  - Does the system store and retrieve information? If so, which actors trigger this behavior?
  - Are any actors notified when the system changes state?
  - Are there any external events that affect the system? What notifies the system about those events?

确定用例 cont’d

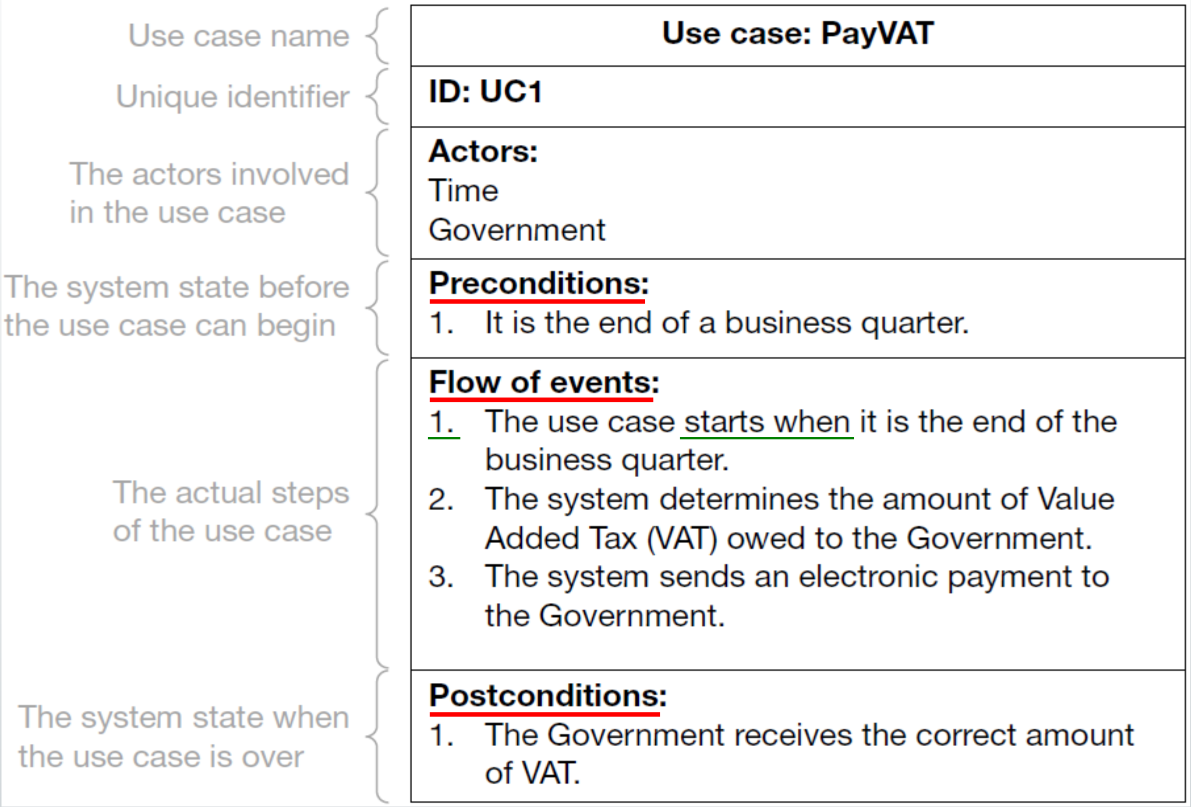
- 用例图



确定项目术语 Project Glossary

- **提供关键商业词汇、定义字典，并被所有参与者所理解**
- **Avoid**
  - Synonyms: different words that mean the same thing
  - Homonyms: the same word means different things to different people
- **UML does not set any standards for a Project Glossary.**
- **Most UML CASE tools do not provide any support for this and so it is usually a manual activity.**

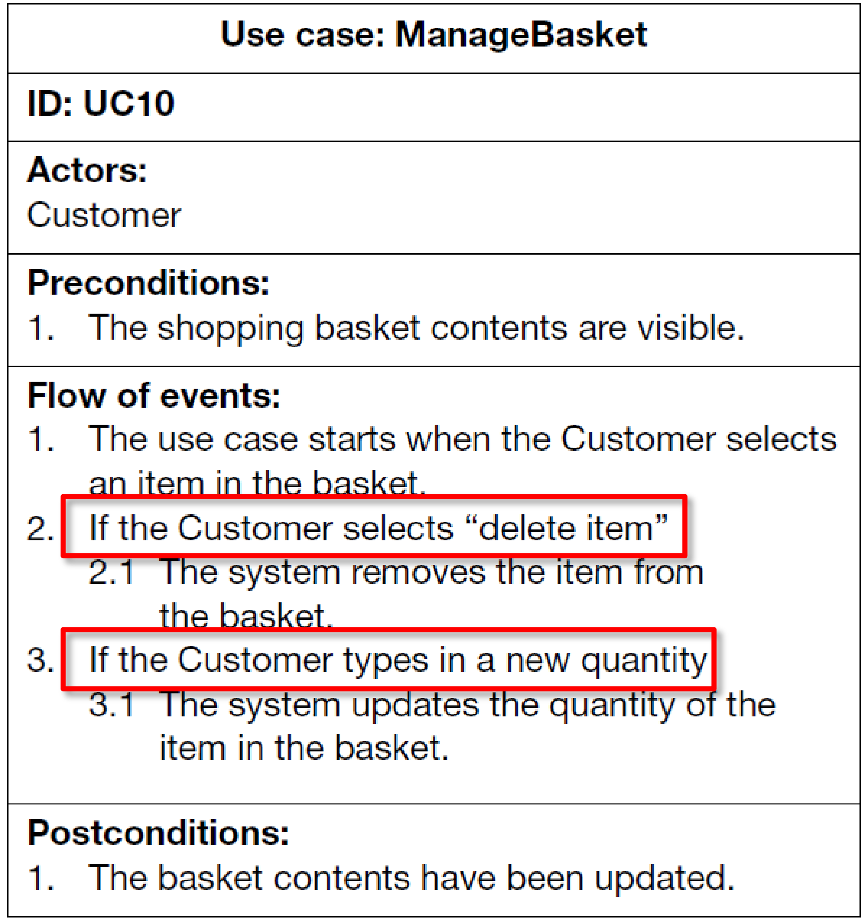
规划描述用例



Use case name	**Use case: PayVAT**
Unique identifier	**ID: UC1**
The actors involved in the use case	**Actors:** Time Government
The system state before the use case can begin	**Preconditions:** 1. It is the end of a business quarter.
The actual steps of the use case	**Flow of events:** 1. The use case starts when it is the end of the business quarter. 2. The system determines the amount of Value Added Tax (VAT) owed to the Government. 3. The system sends an electronic payment to the Government.
The system state when the use case is over	**Postconditions:** 1. The Government receives the correct amount of VAT.

规范描述用例 cont’d

□ Branching within a flow: **If**



Use case: ManageBasket

ID: UC10

Actors:

Customer

Preconditions:

1. The shopping basket contents are visible.

Flow of events:

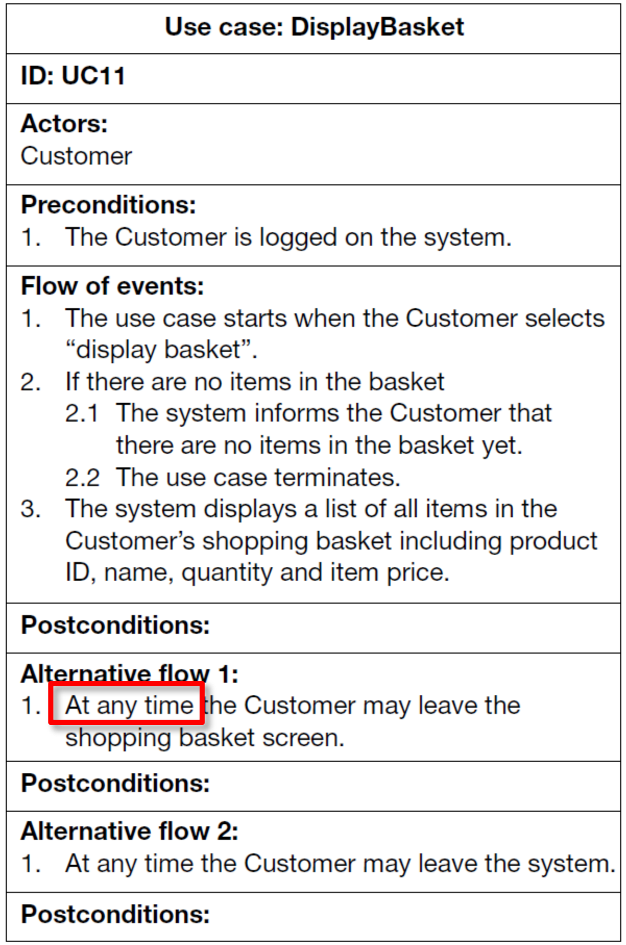
1. The use case starts when the Customer selects an item in the basket.
2. If the Customer selects “delete item”
   - 2.1 The system removes the item from the basket.
3. If the Customer types in a new quantity
   - 3.1 The system updates the quantity of the item in the basket.

Postconditions:

1. The basket contents have been updated.

规范描述用例, cont’d

- **Alternative Flow**
  - Sometimes you can’t easily express branching This is particularly true for things that can happen at any point in time.



Use case: DisplayBasket

ID: UC11

Actors:
Customer

Preconditions:
1. The Customer is logged on the system.

Flow of events:
1. The use case starts when the Customer selects “display basket”.
2. If there are no items in the basket
   2.1 The system informs the Customer that there are no items in the basket yet.
   2.2 The use case terminates.
3. The system displays a list of all items in the Customer’s shopping basket including product ID, name, quantity and item price.

Postconditions:

Alternative flow 1:
1. At any time the Customer may leave the shopping basket screen.

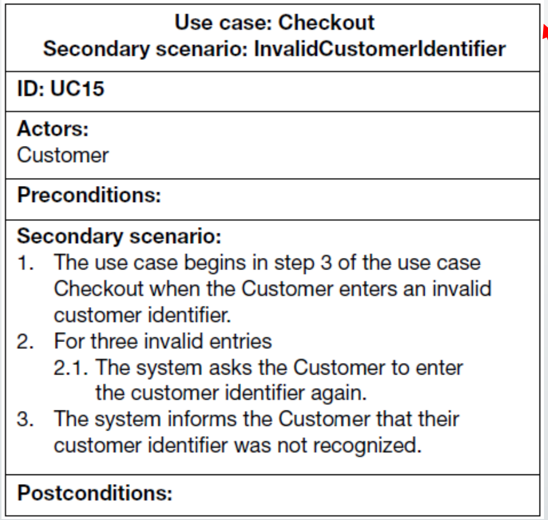
Postconditions:

Alternative flow 2:
1. At any time the Customer may leave the system.

Postconditions:

规范描述用例, cont’d

- Multiple Scenarios



Use case: Checkout
Secondary scenario: InvalidCustomerIdentifier

ID: UC15

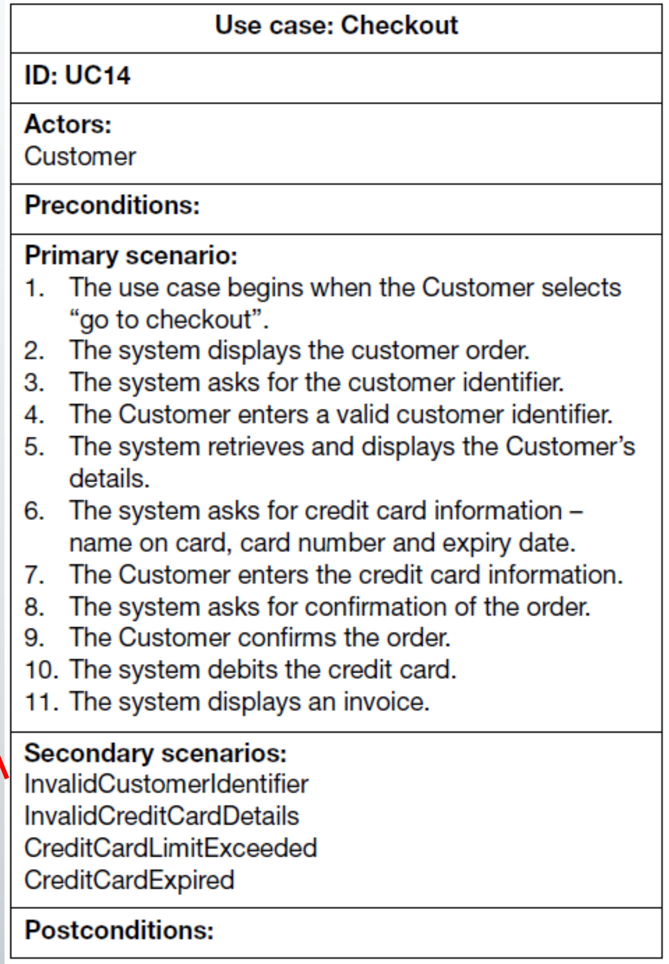
Actors:
Customer

Preconditions:

Secondary scenario:
1. The use case begins in step 3 of the use case Checkout when the Customer enters an invalid customer identifier.
2. For three invalid entries
   2.1. The system asks the Customer to enter the customer identifier again.
3. The system informs the Customer that their customer identifier was not recognized.

Postconditions:

How to find secondary scenarios:
- possible alternative flows;
- errors that might be raised;
- interrupts that might occur to the flow – things that might happen at any time.



Use case: Checkout

ID: UC14

Actors:
Customer

Preconditions:

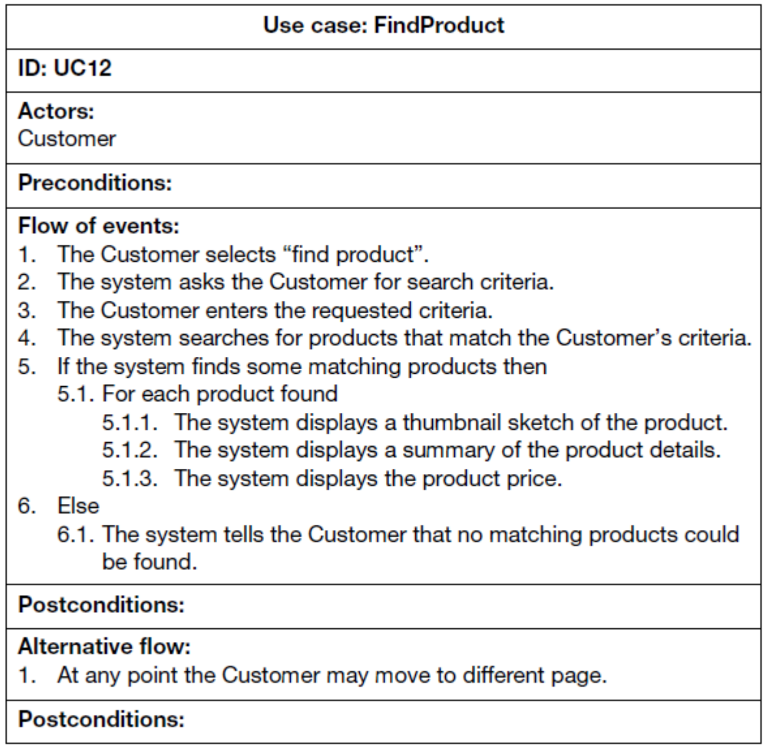
Primary scenario:
1. The use case begins when the Customer selects “go to checkout”.
2. The system displays the customer order.
3. The system asks for the customer identifier.
4. The Customer enters a valid customer identifier.
5. The system retrieves and displays the Customer’s details.
6. The system asks for credit card information – name on card, card number and expiry date.
7. The Customer enters the credit card information.
8. The system asks for confirmation of the order.
9. The Customer confirms the order.
10. The system debits the credit card.
11. The system displays an invoice.

Secondary scenarios:
InvalidCustomerIdentifier
InvalidCreditCardDetails
CreditCardLimitExceeded
CreditCardExpired

Postconditions:

规范描述用例, cont’d

□ Repetition within a flow: For and While



Use case: FindProduct

ID: UC12

Actors:
Customer

Preconditions:

Flow of events:

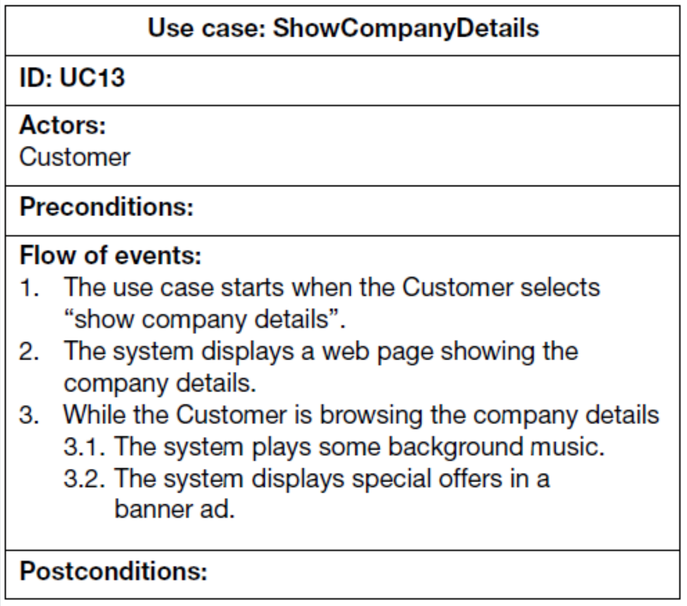
1. The Customer selects “find product”.
2. The system asks the Customer for search criteria.
3. The Customer enters the requested criteria.
4. The system searches for products that match the Customer’s criteria.
5. If the system finds some matching products then
   5.1. For each product found
      5.1.1. The system displays a thumbnail sketch of the product.
      5.1.2. The system displays a summary of the product details.
      5.1.3. The system displays the product price.
6. Else
   6.1. The system tells the Customer that no matching products could be found.

Postconditions:

Alternative flow:

1. At any point the Customer may move to different page.

Postconditions:



Use case: ShowCompanyDetails

ID: UC13

Actors:
Customer

Preconditions:

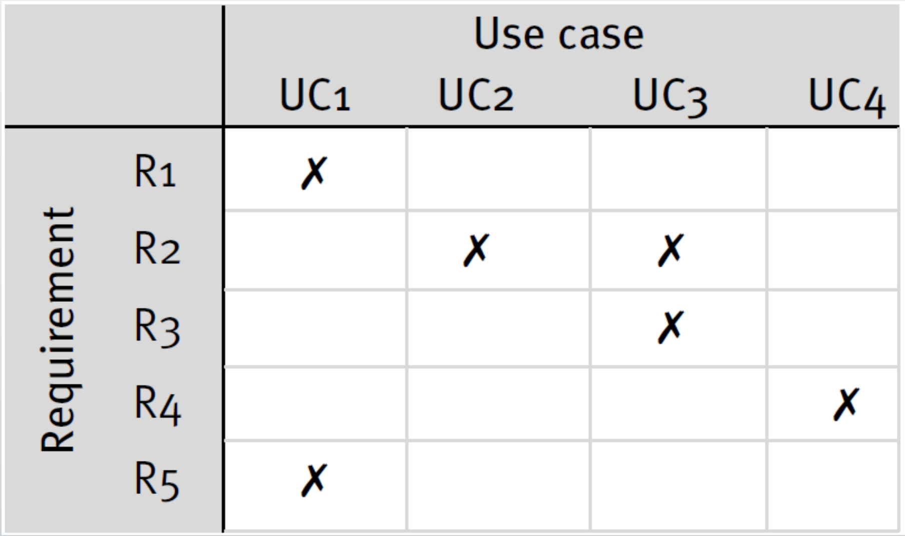
Flow of events:

1. The use case starts when the Customer selects “show company details”.
2. The system displays a web page showing the company details.
3. While the Customer is browsing the company details
   3.1. The system plays some background music.
   3.2. The system displays special offers in a banner ad.

Postconditions:

需求追踪

- **Now we have SRS and use case diagrams**
- **实现需求与用例的匹配**
- **Tools:**
  - CASE tools, e.g., Rational Requisite Pro.
  - Manually by a Requirements Traceability Matrix



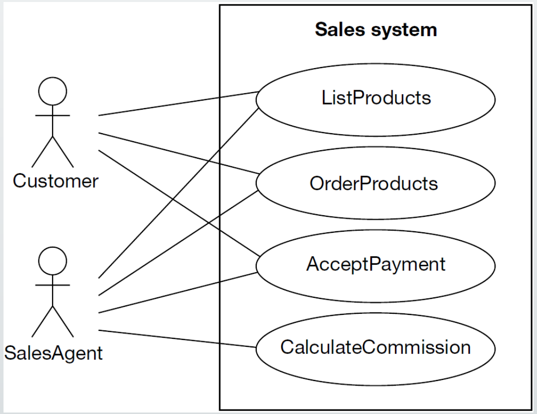
		Use case			
		UC1	UC2	UC3	UC4
Requirement	R1	✗			
	R2		✗	✗	
	R3			✗	
	R4				✗
	R5	✗			

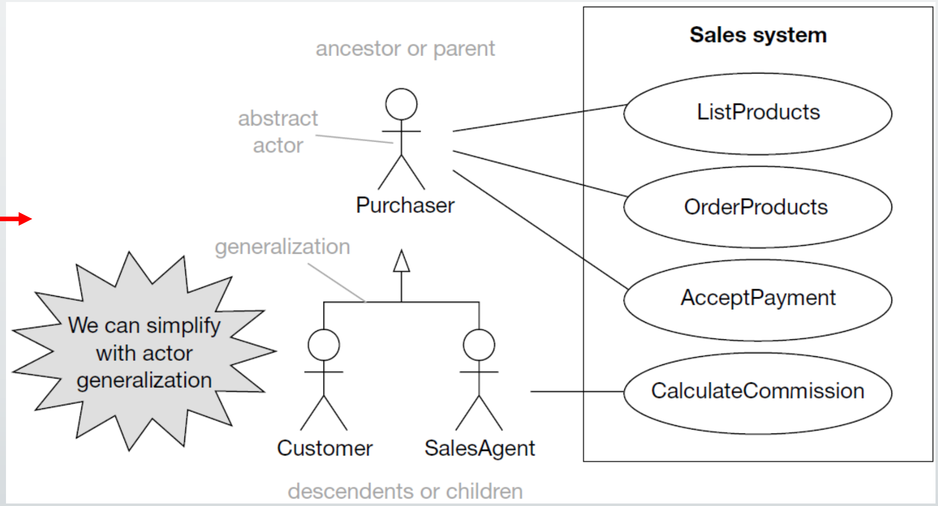
用例建模原则

- **To keep the amount of information captured to the necessary minimum.**
  - To understand the desired behavior of the system, and not just for the sake of creating a complete use case model.
  - 同一建模过程是一个迭代过程
- **使用用例建模的条件**
  - The system is dominated by functional requirements;
  - The system has many types of users to which it delivers different functionality (there are many actors);
  - The system has many interfaces (there are many actors).

参与者泛化：Actor Generalization

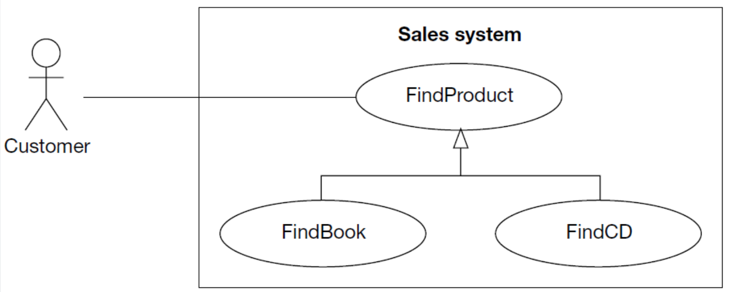
- **Actor generalization factors out behavior common to two or more actors into a parent actor.**



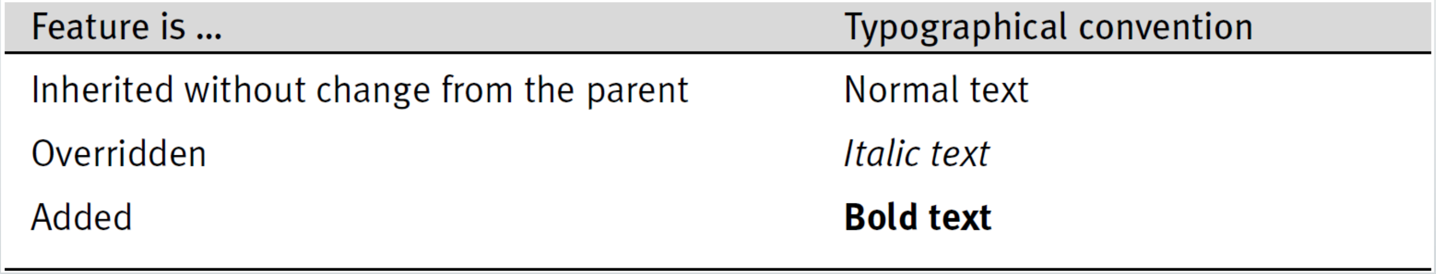


用例泛化：Use Case Generalization

- **Use case generalization factors out behavior common to one or more use cases into a parent use case.**



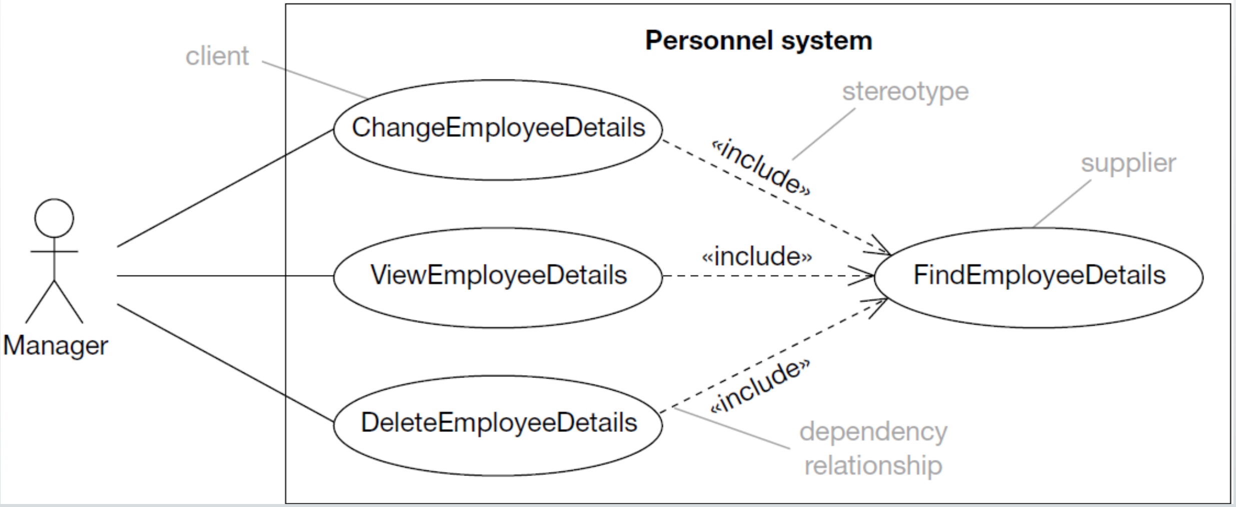
Details can be found on P. 83-84, En. Textbook.



Feature is ...	Typographical convention
Inherited without change from the parent	Normal text
Overridden	*Italic text*
Added	**Bold text**

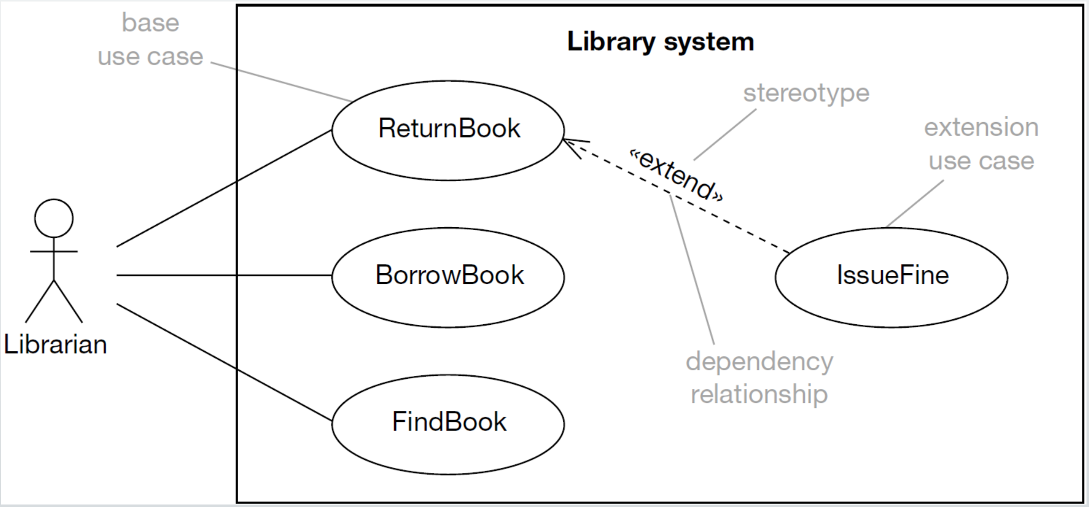
<<include>> Relationship

- «include» factors out steps common to several use cases into a separate use case which is then included.
- 也可称为<<uses>>

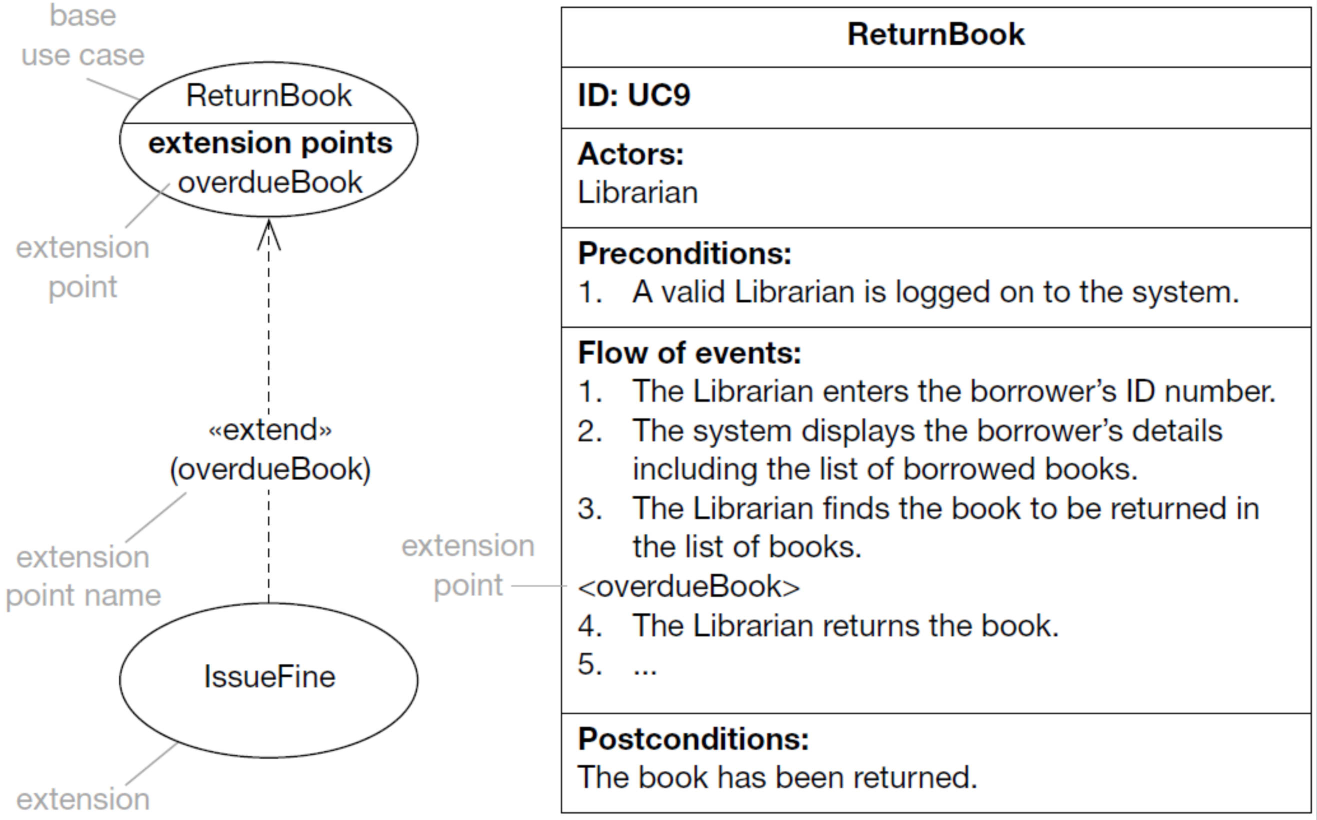


<<extend>> Relationship

- <<extend>> is a way of **inserting new behavior** into an existing use case.
- 扩展用例（extension use case）只能被它扩展的用例（base use case）调用



<<extend>> Relationship, cont'd



ReturnBook

ID: UC9

Actors:
Librarian

Preconditions:
1. A valid Librarian is logged on to the system.

Flow of events:
1. The Librarian enters the borrower's ID number.
2. The system displays the borrower's details including the list of borrowed books.
3. The Librarian finds the book to be returned in the list of books.

<overdueBook>

4. The Librarian returns the book.
5. ...

Postconditions:
The book has been returned.

使用高级用例的原则

- **Use advanced features only when they simplify the model and make it easier to understand:**
  - Stakeholders find actor generalization more difficult to grasp.
  - Use case generalization should be avoided unless **abstract (rather than concrete) parent** use cases are used.
  - Heavy use of «include» can make use case models harder to understand.
  - Stakeholders have great difficulty with «extend». A surprising number of object modelers even misunderstand the semantics of «extend».

用例的本质特征

- **用例是相对独立的**
  - 系统是一个封闭的、由一系列相关关联、相互影响的物质构成的集合
  - 封闭性带来了用例的独立性
  - 从功能上是完备的
- **不存在没有参与者的用例，用例不应该自动启动**
  - 系统随时处于动态，物质的静态积累不可能构成系统
- **用例的执行结果对参与者来说是可观测和有意义的**
  - 只有有效的反馈才能使系统保持动态平衡状态
- **用例必然是以动宾短语的形式出现**
  - 动作和动作的受体

用例粒度

- **粒度**
  - 哪些活动、功能是可以构成用例的？
- **用例粒度划分的依据**
  - 参与者是否完成了某个完整的目的
  - 某人去图书馆，查询书目，出示借书证，图书管理员查询该人历史借阅记录是否有未归还，最后借到了书
- **不同类型的系统和需求范围所对应的系统用例粒度选择不同**
  - 50人年的项目
  - 10人月的项目

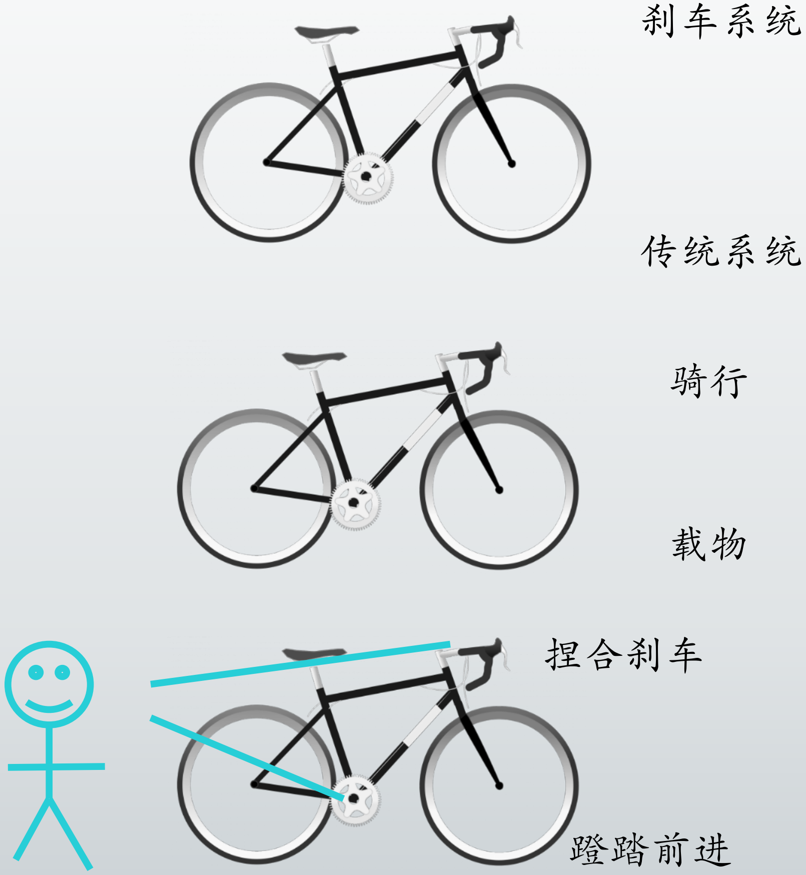
ATM例子

我希望这台ATM能支持跨行业务，我插入卡片输入密码后，可以让我选择取钱还是存钱；为了方便，可以设置默认的存取金额按钮；可以修改密码，也可以挂失卡片；希望可以缴纳电话费、水费、电费等费用；为了安全起见，ATM上应该有警示小心骗子的提示条，还有摄像头；如果输入三次密码错误，卡片应该被自动吞没。

- **支持跨行业务？**
- **插入卡片？**
- **输入密码？**
- **选择服务？**
- **取钱？**
- **存钱？**
- **挂失卡片？**
- **缴纳费用？**
- **警示骗子？**
- **三次错误吞没卡片？**

用例与功能

- **用例是否是功能的划分和描述？**
- **参与者角度出发**
- **描述事物的三个观点**
  - 结构观点：是什么？
  - 功能观点：能做什么？
  - 使用者观点：人们能够用这个事物做什么



刹车系统

传统系统

骑行

载物

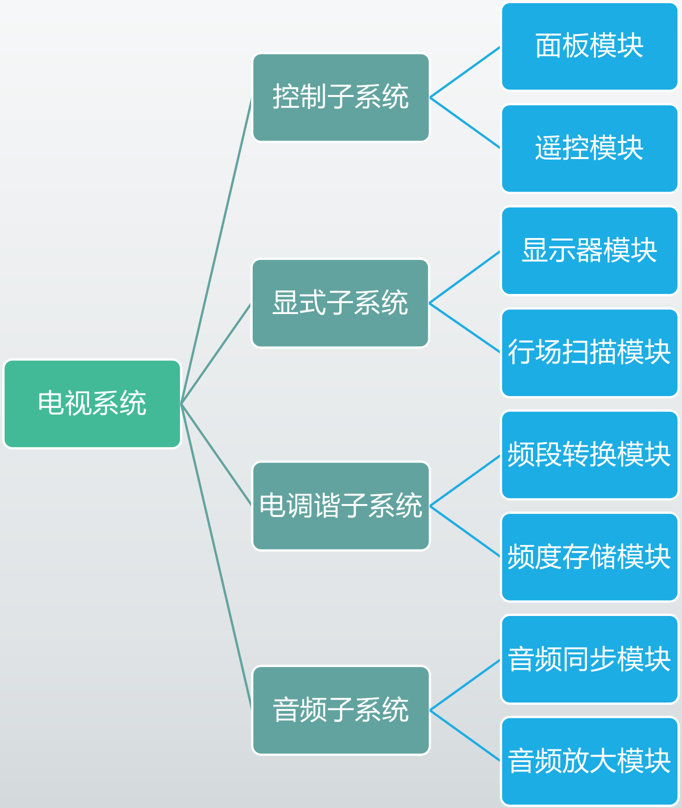
捏合刹车

蹬踏前进

目标、功能、步骤

- **信息系统适合用结构来描述吗？**
  - 还不存在的事物，不能够确定它到底能做什么
- **适合用功能来描述吗？**
- **功能是脱离使用者的愿望而存在的**
  - 描述的事物固有性质
- **功能是孤立的**
  - 描述的是一个个点
- **用例可以认为是一系列完成一个特定目标的“功能”组合**

不同思维下的电视机

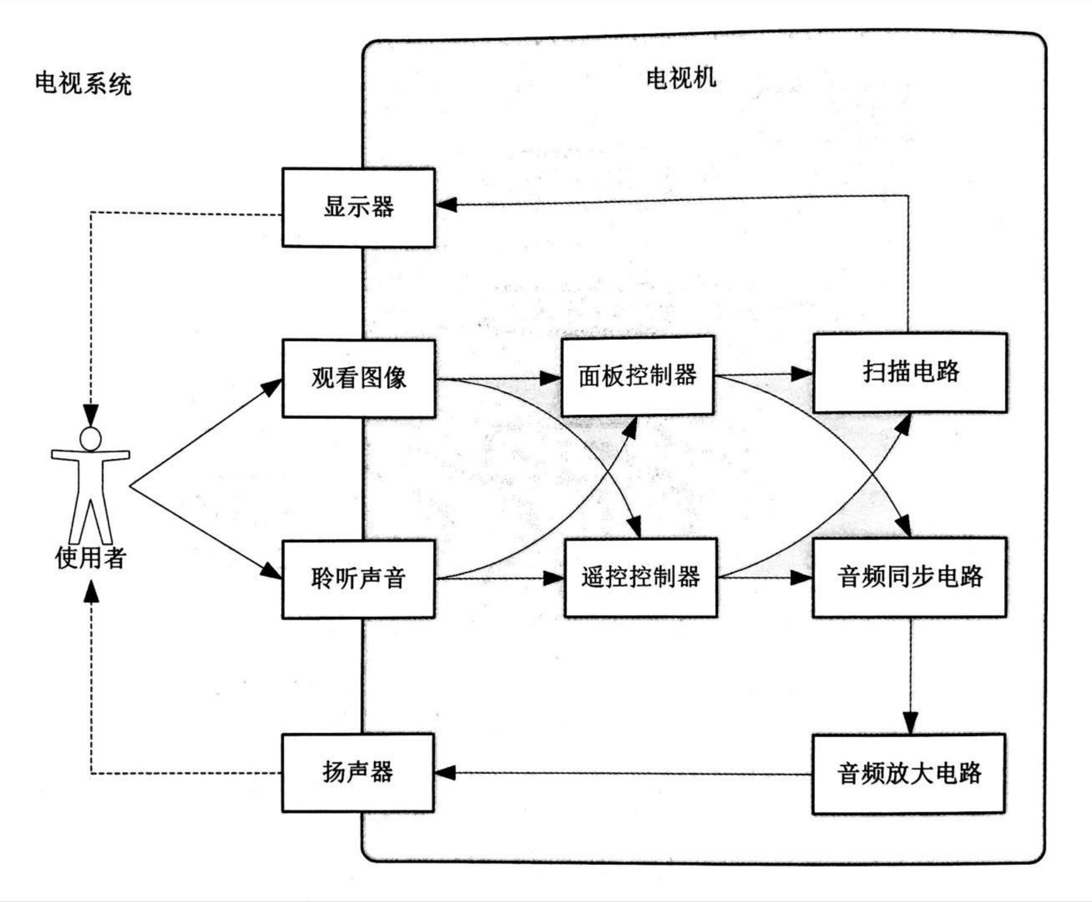


典型的结构化/线性思维

忽略了什么基本事实?

电视机与使用者共同构成系统!

电视的用例？


电视系统
电视机
显示器
观看图像
面板控制器
扫描电路
使用者
聆听声音
遥控控制器
音频同步电路
扬声器
音频放大电路

刘冠男

liugn@buaa.edu.cn

数据智能研究中心
RESEARCH CENTER FOR
DATA INTELLIGENCE