# UML概念

（UML中所有的箭头都表示导航，如果没有方向意义的话可以省略）

## 场景：

* **用户与系统之间的一个交互过程，即为实现这次交互所要经历的一系列步骤**

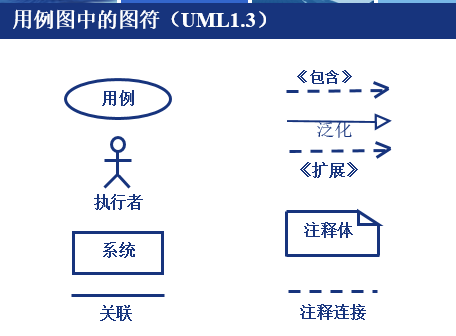
## 用例：

* **一组场景，用以共同描述用户的某个特定的目标。**

## 用例分析的目的

* + **1. 描述和决定系统的功能需求，帮助客户和软件开发人员形成一致意见。**
  + **2. 给出系统应该做什么且与内容一致的可视化描述，使之成为在开发全过程中研讨系统需求和进行系统设计的依据。**
  + **3. 在软件测试阶段作为系统测试的基础。建立系统实现的各个对象类和系统操作与功能需求之间的可追踪关系。**

## 用例分析的特点

* **用例用于描述系统的功能，这个功能是外部使用者看到的系统功能，不反映功能的实现方式**
* **用例描述用户提出的一些可见需求，对应一个具体的用户目标。**
* **用例反映系统与用户的一次交互过程，应该具有交互的信息的传递。**
* 

## 系统边界：

* **一个提供“用例”所需要的功能的“黑盒 子”。系统的外部特性由系统的功能来定义；整个系统的功能用一组用例来描述。**

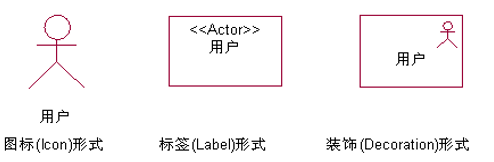
## 执行者（**参与者**）：

* **需要使用系统的任何外部实体(例如：人、其它系统或外部设备等)。**
* **（如果被指，就是动作承受者）**

### 1. 参与者的概念

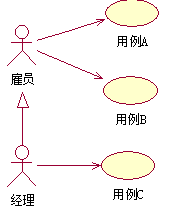
* **参与者（actor）是外部需要与系统交互的事物。也被称为活动者或执行者。**

### 2.参与者的三种类型

* **①. 人：客户，读者，库管员**
* **②. 设备：计算机，磁盘，读卡机等**
* **③. 外部系统：上层系统等**
* 

### 参与者之间的关系

#### 泛化关系

* 

## 用例：

* **用客户或用户的语言和词汇来描述的系统的一个完整功能。**

## 用例间关系

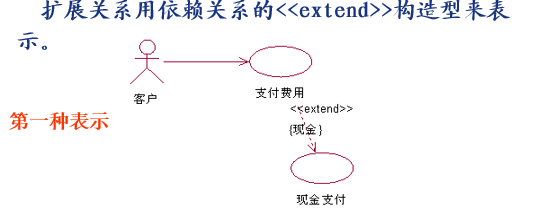
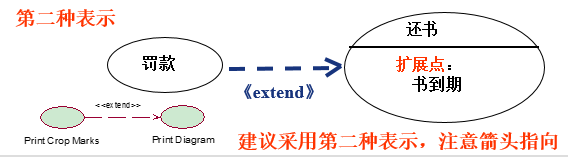
### 关联：

* **连接执行者和用例，表示该执行者所代表的系统外部实体与该用例所描述的系统需求有关。这是执行者和用例之间的唯一合法连接。（无箭头或带箭头实线）**
* **根据用例的发起者不同，把角色和用例间的关联（通信）关系分为单向管理和双向关联**

### 包含（使用）：

* **由用例A连向用例B，表示用例A中使用了用例B中的行为或功能。（虚线尖箭头）（《include》建议用<<uses>>）**

### 扩展：

* **由用例 A连向用例 B，表示用例B（父用例）描述了一项基本需求（比较虚的），而用例A则描述了该基本需求的特殊情况，即一种扩展。箭头所指为父用例。<<extend>>（虚线尖箭头）**
* 
* 

### **泛化**

* **参与者与参与者之间，用例与用例之间存在一般与特殊的关系。一个用例可以特化另一个更普通用例。（空心实线箭头）**
* **(箭头指向基用例)**

# 建立用例模型的主要工作

## 定义系统；

* + - **系统的属性；**
    - **定义系统时的注意点；**
    - **对与外界系统交互问题的看法**

## 找出执行者；

* + - **执行者的主要属性；**
    - **执行者与系统和用例之间的关系；**
    - **执行者的获取；**
    - **对执行者的描述。**

## 找出用例；

* + - **① 找出系统外部参与者，确定系统边界和范围。**
  + **② 确定各参与者所期望的系统行为。**
  + **③ 把这些系统行为命名为用例。**
  + **④ 确定各用例之间的关系(泛化，包含，扩展)。**
  + **⑤ 绘制用例图。**
  + **⑥ 编制用例说明。**
  + **⑦ 对异常流程确定单独用例。**
  + **⑧ 优化用例图，解决用例之间的冲突和重复。**

### Tips

* **首先获取简单的、常规的用例作为基本用例(在上例中，基本用例是进行交易)。**
* **当一个用例与另一个用例相似，但比另一个用例做的动作要多，采用扩展关系：**
* **对用例中的每一步，问一下“这儿可能出现什么异常情况”以及“是否需要采取不同的解决方法”；将所有出现变动的部分列出，作为给定用例的扩展。**
* **一旦获取了系统的执行者，就可对每个执行者提出一些问题，然后从执行者对这些问题的答案中获取用例。**
* **为了完整地描述用例，还需要知道执行者的某些典型功能能否被系统自动实现。**
* **针对整个系统的问题的答案也可帮助我们获取用例。以下问题可供参考：**
* **系统需要何种输入输出？输入从何处来？输出到何处去？**
* **当前运行系统(也许是一些手工操作而不是计算机系统)的主要问题是什么？**
* **在开发系统的用例图时，不同的设计者选取用例的数目也不相同。**

## 描述用例；

## 用例的整理与加工；

## 验证模型。

# 用例说明文档

* **1）前置条件**
* **2）后置条件**
* **3）基本事件流**
* **4）可选事件流**
* **5）异常事件流**
* **6）参与者**
* **7）用例名称**