

# 第3章 需求分析

需求分析的任务

与用户沟通获取需求的方法

分析建模与规格说明

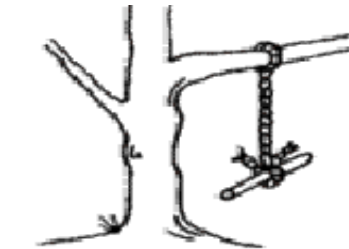
实体-联系图

状态转换图

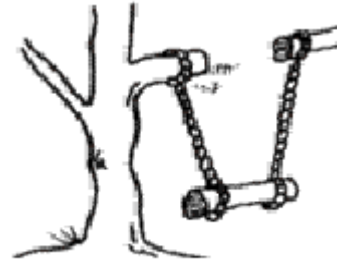
# 第3章 需求分析

- 需求分析是软件定义时期的最后一个阶段，它的基本任务是准确地回答“系统必须做什么？”这个问题。
- 确定系统必须完成哪些工作，也就是对目标系统提出完整、准确、清晰、具体的要求。
- 系统分析员应该写出软件需求规格说明书，以书面形式准确地描述软件需求。

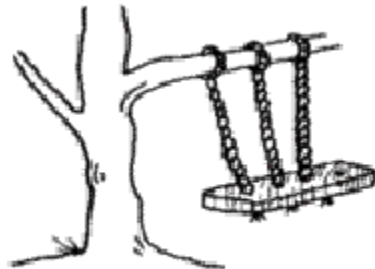
# 需求管理的困难性



What the user asked for



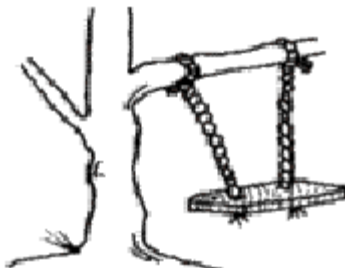
How the analyst saw it



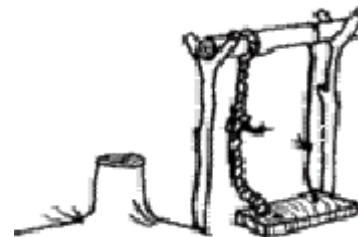
How the system was designed



As the programmer wrote it



What the user really wanted



How it actually works

# 3.1 需求分析的任务

- 确定对系统的综合要求
- 分析系统的数据要求
- 导出系统的逻辑模型
- 修正系统开发计划

# 3.1.1 确定对系统的综合要求

1. 功能需求
2. 性能需求
3. 可靠性和可用性需求
4. 出错处理需求
5. 接口需求
6. 约束
7. 逆向需求
8. 将来可能提出的要求

## 3.1.2 分析系统的数据要求

- 建立数据模型——**ER图**
- 描绘数据结构——层次方框图和**Warnier图**
- 数据结构规范化

## 3.1.3 导出系统的逻辑模型

- 综合上述两项分析的结果可以导出系统的详细的逻辑模型，通常用数据流图、实体-联系图、状态转换图、数据字典和主要的处理算法描述这个逻辑模型。

## 3.1.4 修正系统开发计划

- 根据在分析过程中获得的对系统的更深入更具体的了解，可以比较准确地估计系统的成本和进度，修正以前制定的开发计划。



# 需求获取的来源

- 1) 用户访谈与会议;
- 2) 把对目前的或竞争产品的描述写成文档;
- 3) 系统需求规格说明;
- 4) 对当前系统的问题报告和增强要求;
- 5) 市场调查和用户问卷调查;
- 6) 观察用户工作流程;
- 7) 用户任务内容分析;
- 8) 用户与开发人员共同组成联合小组

## 3.3 分析建模与规格说明

### 3.3.1 分析建模

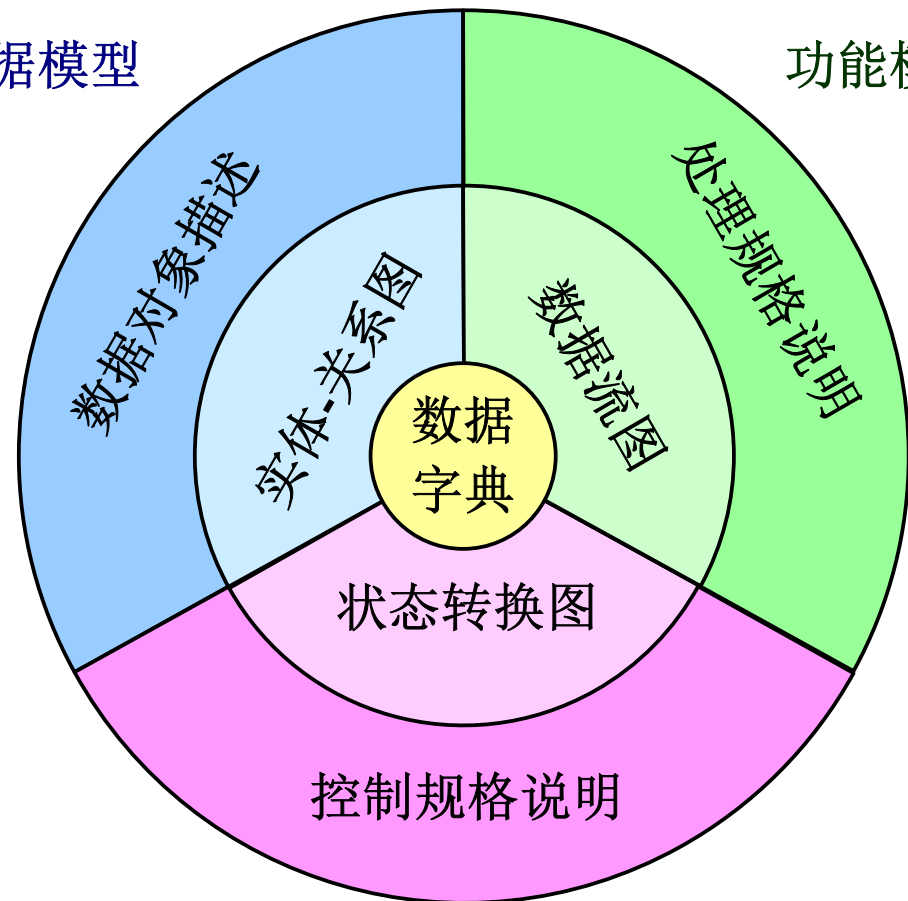
- **模型：**就是为了理解事物而对事物做出的一种抽象，是对事物的一种无歧义的书面描述。通常，模型由一组图形符号和组织这些符号的规则组成。
- **结构化分析过程：**实质上是一种创建模型的活动。系统分析员从不同角度抽象出目标系统的特性，使用精确的表示方法构造系统的模型，验证模型是否满足用户对目标系统的需求，并在设计过程中逐渐把和实现有关的细节加进模型中，直至最终用程序实现模型。

需求分析过程  
应该建立3种模型，分别是：

- 数据模型
- 功能模型
- 行为模型

数据模型

功能模型



行为模型

分析模型的结构

## 3.3.2 软件需求规格说明

- 通过需求分析除了创建分析模型之外，还应该写出软件需求规格说明书，它是需求分析阶段得出的最主要的文档。
- 通常用自然语言完整、准确、具体地描述系统的数据要求、功能需求、性能需求、可靠性和可用性要求、出错处理需求、接口需求、约束、逆向需求以及将来可能提出的要求。

# 我国定义了GB856D-1988国家标准，给出了需求规格说明的内容框架：□

- 1 引言
  - 1.1 编写目的
  - 1.2 项目背景（单位和其他系统的关系）
  - 1.3 定义（专门术语和缩写词）
- 2 任务概述
  - 2.1 目标
  - 2.2 运行环境
  - 2.3 条件限制
- 3 数据描述
  - 3.1 静态数据
  - 3.2 动态数据
  - 3.3 数据库描述
  - 3.4 数据字典
  - 3.5 数据采集
- 4 功能需求
  - 4.1 功能划分
  - 4.2 功能描述
- 5 性能需求
  - 5.1 数据精确度
  - 5.2 时间特性
  - 5.3 适应性
- 6 运行需求
  - 6.1 用户界面
  - 6.2 硬件接口
  - 6.3 软件接口
  - 6.4 故障处理
- 7 其他需求
  - （检测或验收标准、可用性、可维护性、可移植性、安全保密性）

## 3.4 实体-联系图

- 概念性数据模型是一种面向问题的数据模型，是按照用户的观点对数据建立的模型。它描述了从用户角度看到的数据，它反映了用户的现实环境，且与在软件系统中的实现方法无关。
- 数据模型中包含3种相互关联的信息：
  - 数据对象
  - 数据对象的属性
  - 数据对象彼此间相互连接的关系

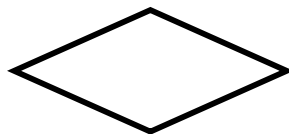
# 实体-联系图的符号

**ER图中包含：**

- 实体(即数据对象)，用矩形框表示；



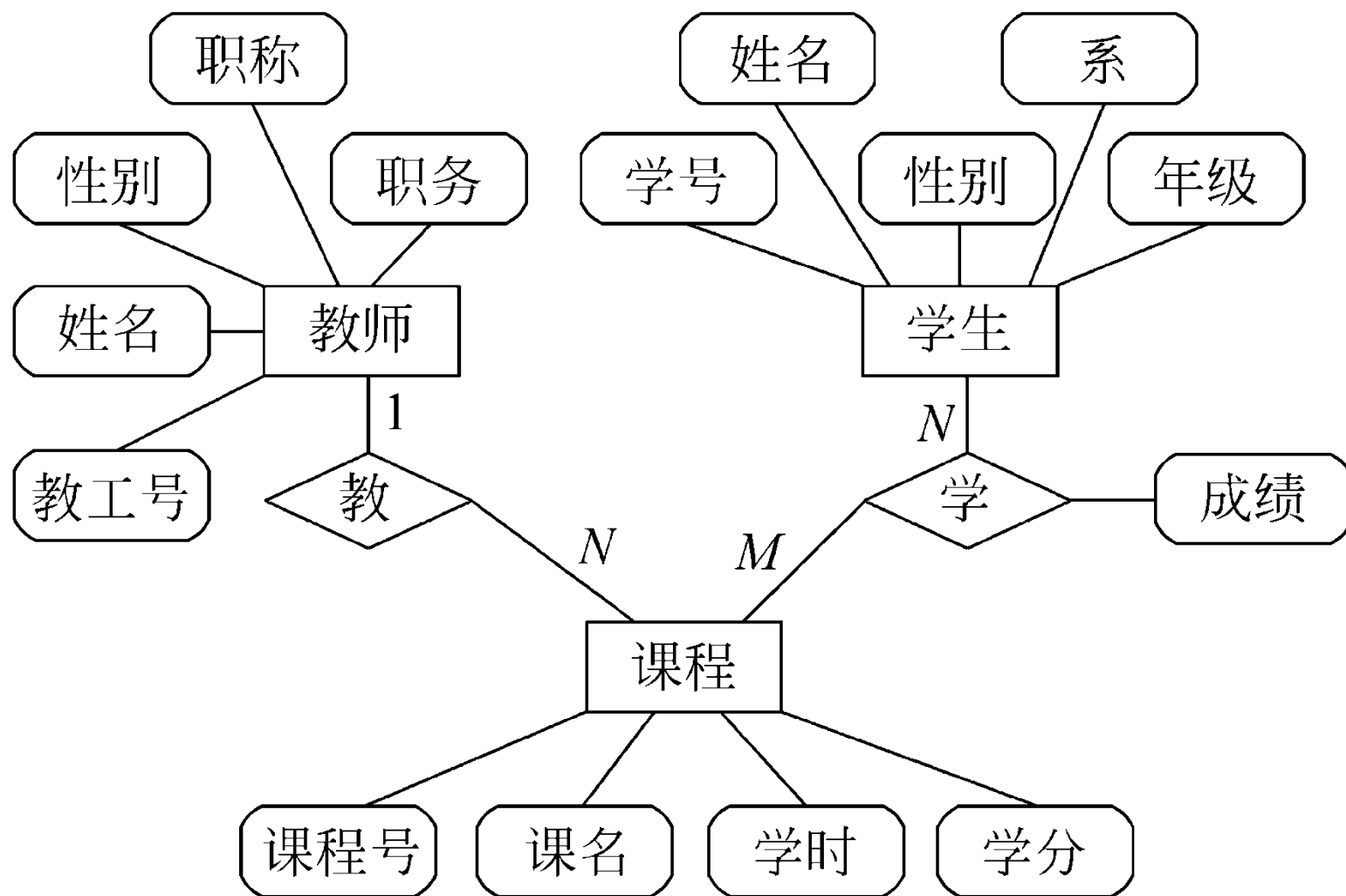
- 关系，用连接相关实体的菱形框表示；



- 属性，用椭圆形或圆角矩形表示，并用直线把实体(或关系)与其属性连接起来。



# 例1：某校教学管理系统的ER图





# 状态转换图

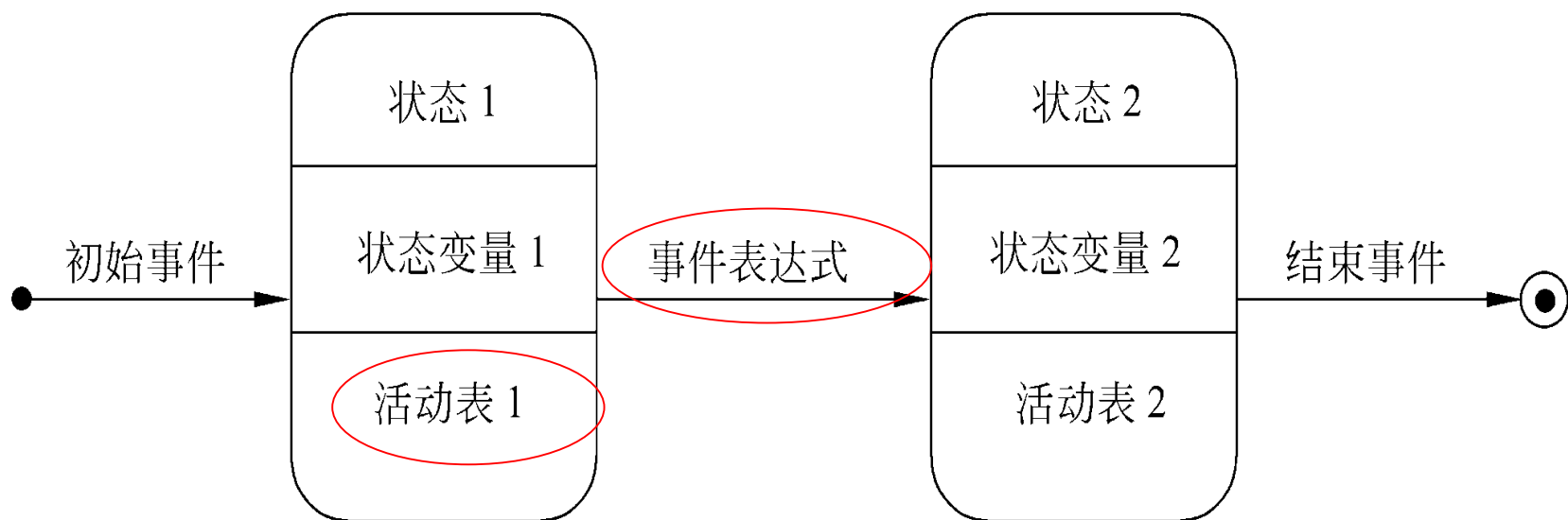
- **状态转换图：**通过描绘系统的状态及引起系统状态转换的事件，来表示系统的行为。
- 状态图还指明了作为特定事件的结果系统将做哪些动作(例如，处理数据)。

## 3.6.1 状态

- **状态**： 是任何可以被观察到的系统行为模式，一个状态代表系统的一种行为模式。状态规定了系统对事件的响应方式。
- **状态主要有**：
  - 初态(即初始状态)，只能有1个
  - 终态(即最终状态)，可以有0至多个
  - 中间状态
- **状态图分类**：
  - 表示系统循环运行过程，通常不关心循环是怎样启动的。
  - 表示系统单程生命期，需要标明初始状态和最终状态。

## 3.6.3 符号

- 初态：用实心圆表示；
- 终态：用一对同心圆(内圆为实心圆)表示；
- 中间状态：用圆角矩形表示，分成上、中、下3部分。
  - 上面部分-----为状态的名称；
  - 中间部分-----为状态变量的名字和值；
  - 下面部分-----是活动表。
- 带箭头的连线：称为状态转换，箭头指明了转换方向。



## 状态图中使用的主要符号

## 活动表的语法格式:

### 事件名(参数表)/动作表达式

- “事件名”可以是任何事件的名称。
- 常用的3种标准事件:
  - **entry**事件指定进入该状态的动作;
  - **exit**事件指定退出该状态的动作;
  - **do**事件则指定在该状态下的动作。
- 需要时可以为事件指定参数表。活动表中的动作表达式描述应做的具体动作。

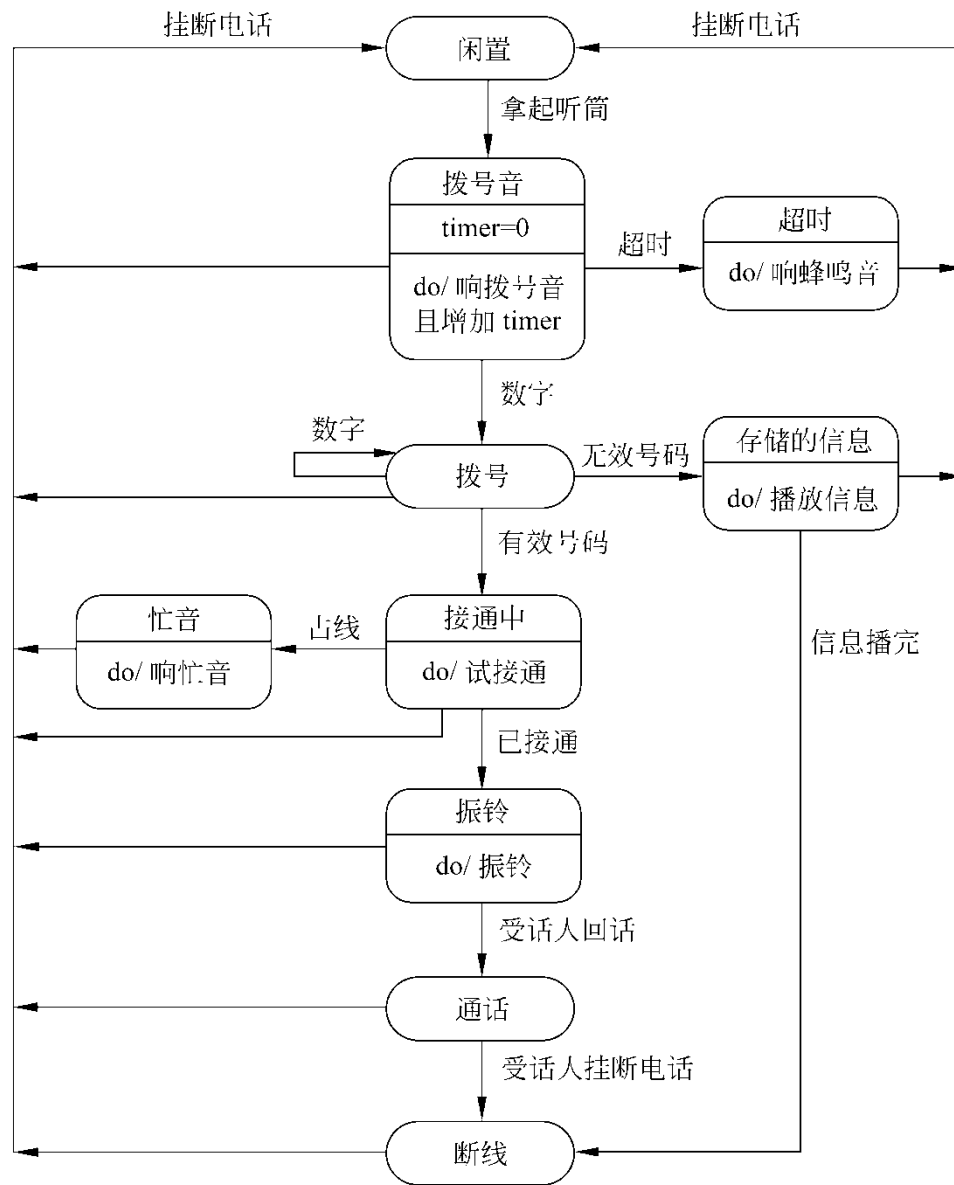
## 事件表达式的语法:

### 事件说明 [ 守卫条件 ] / 动作表达式

- 事件说明的语法为: 事件名(参数表)。
- 守卫条件是一个布尔表达式。如果同时使用事件说明和守卫条件, 则当且仅当事件发生且布尔表达式为真时, 状态转换才发生。如果只有守卫条件没有事件说明, 则只要守卫条件为真状态转换就发生。
- 动作表达式是一个过程表达式, 当状态转换开始时执行该表达式。

## 3.6.4 例子

### 例1：电话系统

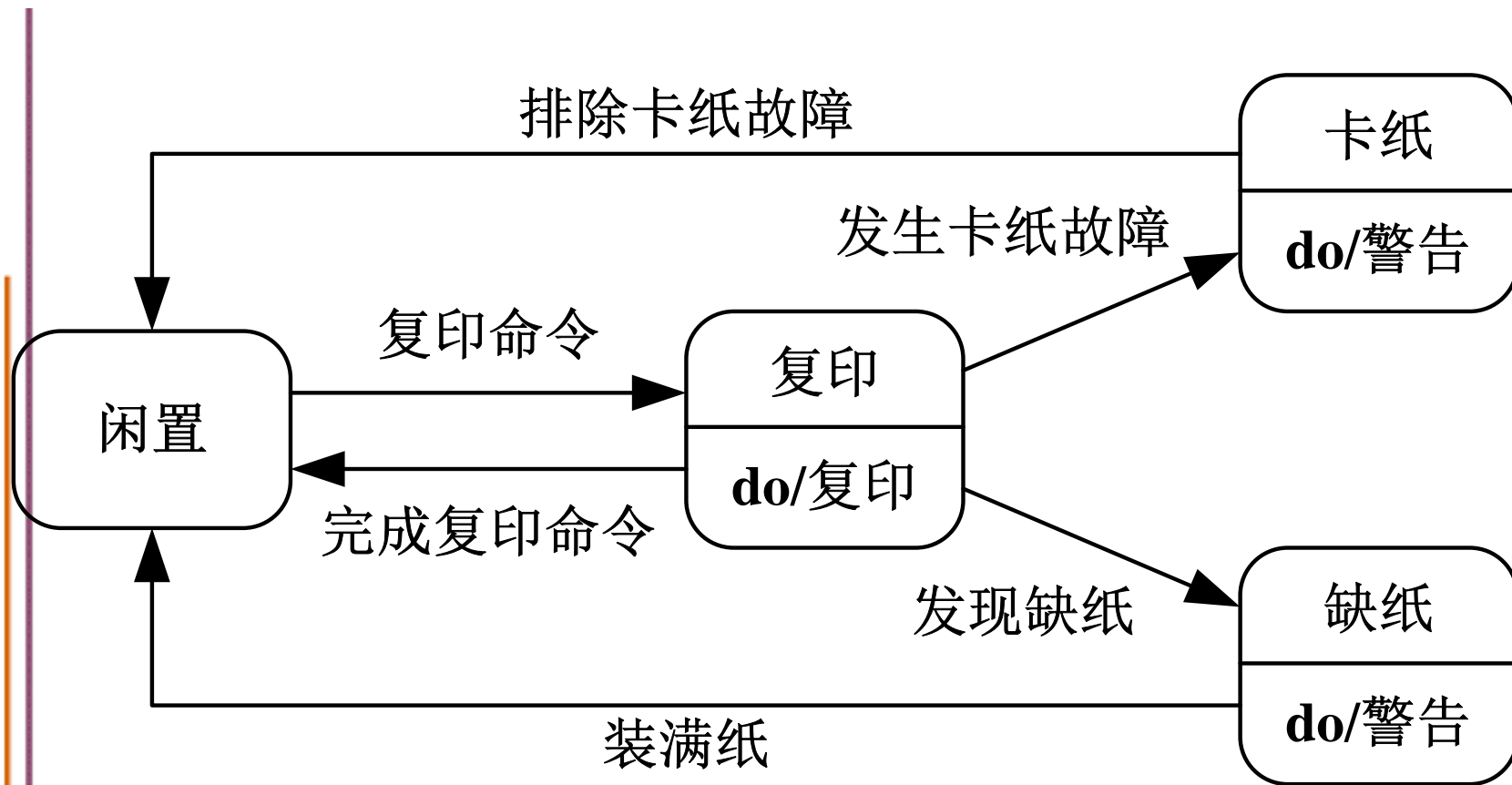


## 状态图练习题：

复印机的工作过程大致如下：

- 未接到复印命令时处于闲置状态，一旦接到复印命令则进入复印状态，完成一个复印命令规定的工作后又回到闲置状态，等待下一个复印命令；
- 如果执行复印命令时发现没纸，则进入缺纸状态，发出警告，等待装纸，装满纸后进入闲置状态，准备接收复印命令；
- 如果复印时发生卡纸故障，则进入卡纸状态，发出警告等待维修人员来排除故障，故障排除后回到闲置状态。





# 小结

---

- 需求分析的任务
- 与用户沟通获取需求的方法
- 分析建模与规格说明
- 实体-联系图
- 状态转换图