



## 第六章 结构化分析方法

---



# 一、用哲学的观点看面向过程和面向对象

## 1、世界观

世界观：是人们对世界的总的根本的看法。任何哲学问题的探讨，归其出发点和本源，都是世界观的问题。什么样的世界观决定了什么样的哲学观点。——马克思”

---





唯物主义<—根本对立——>唯心主义。

在软件开发设计方面不同的人也有着不同的世界观。而这其中最根本的对立便是过程论和对象论的对立，这个对立，衍生出了面向过程和面向对象两种方法论

注意：过程论和对象论，都承认程序世界本质上只有两种东西——数据和逻辑。数据天性喜静，构成了程序世界的本体和状态；逻辑天性好动，作用于数据，推动程序世界的演进和发展。

---



## 过程论观点:

1> 数据和逻辑是分离的、独立的，各自形成程序世界的一个方面。所谓世界的演变，是在逻辑作用下，数据做改变的一个过程。

2> 过程有明确的开始、结束、输入、输出，每个步骤有着严格的因果关系。

3> 过程是相对稳定的、明确的和预定义的，小过程组合成大过程，大过程还可以组合成更大的过程。

**结论：**程序世界本质是过程，数据作为过程处理对象，逻辑作为过程的形式定义，世界就是各个过程不断进行的总体。

---





## 对象论观点:

- 1> 数据和逻辑不是分离的，而是相互依存的。
- 2> 相关的数据和逻辑形成个体，这些个体叫做对象，世界就是由一个个对象组成的。  
相关的数据和逻辑组合成对象
- 3> 对象具有相对独立性， 对外提供一定的服务。
- 4> 所谓世界的演进，是在某个“初始作用力”作用下，对象间相互调用而完成的交互，在没有初始作用力下，对象保持静止。
- 5> 交互并不是完全预定义的，不一定有严格的因果关系，对象间交互是偶然的，对象间联系是暂时的。

**结论：**世界就是由各色对象组成，然后在初始作用力下，对象间的交互 完成了世界的演进。

---



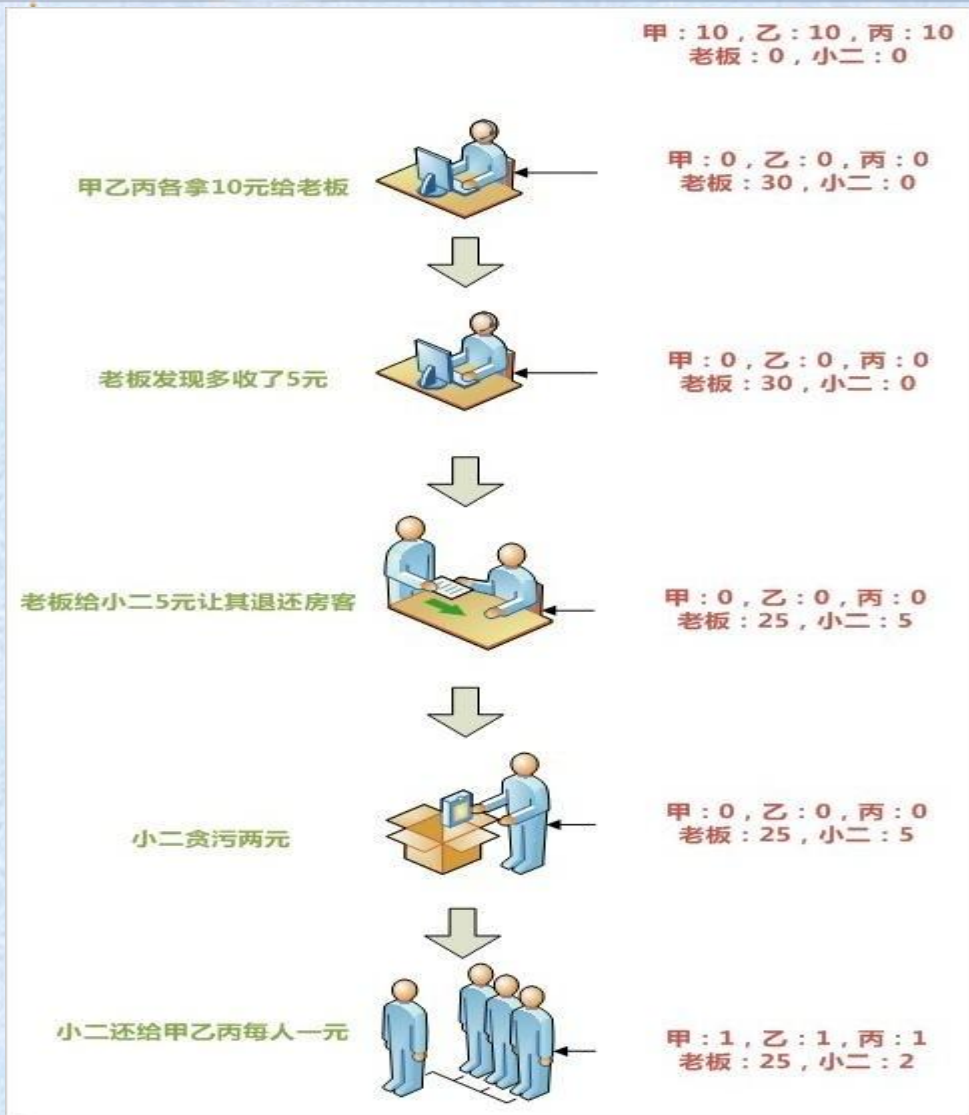


问题：

有甲、乙、丙三人住店，一间房30。于是每人10元，共计给店老板30元住进一间房。后来店老板发现弄错了，房价应该是25元，于是给小二5元让小二退给房客。小二黑心，贪污了2元，退给甲乙丙每人1元。这样房客每人付了 $10-1=9$ 元，三九27，加上小二贪污的2元，共29元，问那1元哪里去了？

如何看待和解决上面的问题？

---



经历五个步骤  
后，数据变为  
甲乙丙各1元，  
老板25元，小  
二2元，这就是  
终止状态，也  
是这个过程的  
输出

对应考虑结构  
化分析的模型

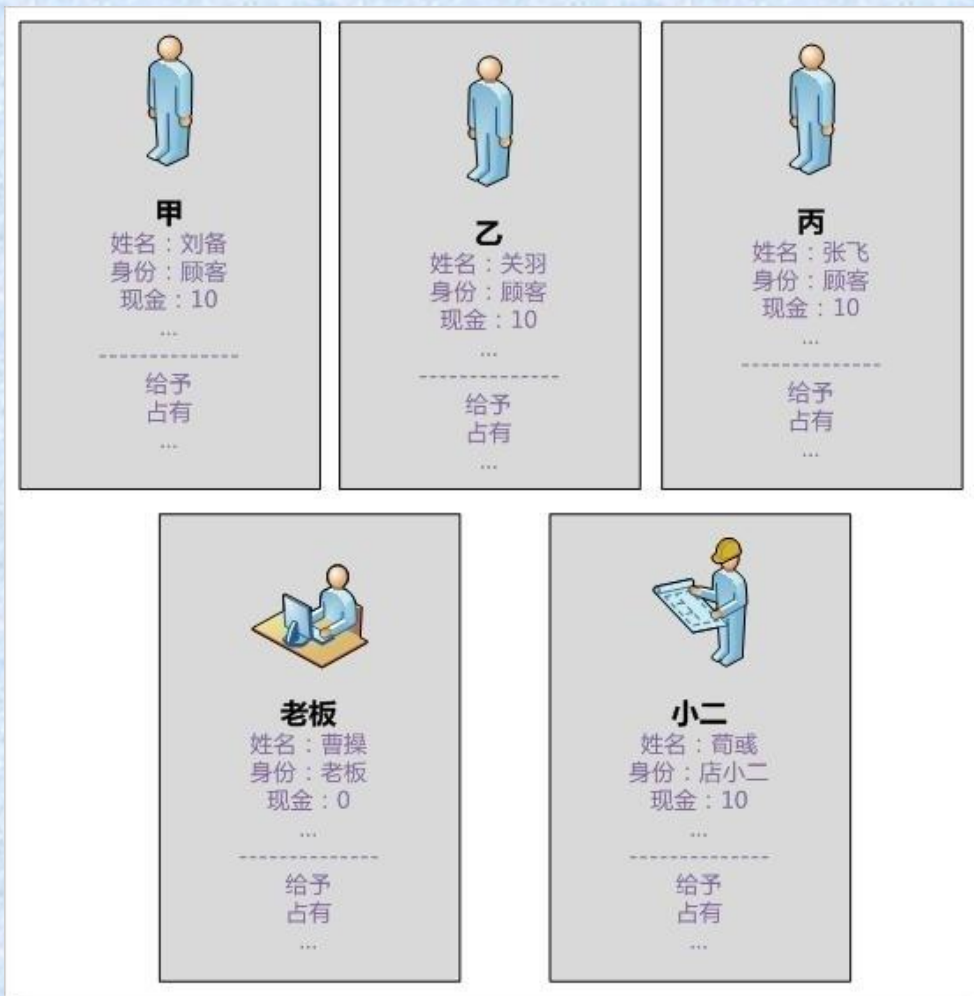




对象论眼中，有五个基本对象，每个对象有自己的一系列数据和逻辑没错，在对象论眼里，这就是这件事的本质模样，这件事所涉及的东西就是这么几个对象。

本来它们各自独立，老死不相往来。只不过在住店这个外部驱动力下，几个对象偶然、暂时互相联系，利用其他对象提供的公开服务，完成了一些交互。

对应考虑面向对象分析的模型



## 对象论观点





## 两种思想的比较：

- 过程论和对象论是两种看世界的观点，没有好坏之分。
  - 过程论和对象论不是绝对对立，而是一种辩证统一的对立，两者相互渗透、在一定情况下可以相互转化。如果将对象论中的所有交互提取出来而撇开对象，就变成了过程论，而如果对过程论中的数据和逻辑分类封装并建立交互关系，就变成了对象论。
  - 过程论相对确定，有利于明晰演进的方向，但当事物过于庞大繁杂，将很难理清思路。因为过程繁多、过程中又有子过程，容易将整个世界看成一个纷繁交错的过程网，让人无法看清。
  - 对象论相对不确定，但是因为以对象为基本元素，即使很庞大的事物，也可以很好地分离关注，在研究一个对象的交互时，只需要关系与其相关的少数几个对象，不用总是关注整个流程和世界。但是，对象论也有困难。例如，如何划分对象才合理？对于同一个驱动力，为什么不同情况下参与对象和交互流程不一样？如何确定？其实，这些困难也正是面向对象技术中的困难。
-





## 二、结构化分析方法概述

### 1、什么是结构化分析方法？

- 结构化分析方法是一种建模技术
  - 基于计算机的系统是数据流和一系列的转换构成的
  - 在模型的核心是数据词典，它描述了所有的在目标系统中使用的和生成的数据对象。围绕着这个核心的有三种图： ERD、DFD、STD
-





## 2、结构化分析模型的组成

- 数据建模和对象描述
  - 功能建模和数据流图
  - 基本加工逻辑说明
  - 行为建模
  - 数据词典
-



### 3、基本思想

基本思想：抽象与自顶向下的逐层分解  
(控制复杂性的两个基本手段)

**抽象：** 在每个抽象层次上忽略问题的内部复杂性，只关注整个问题与外界的联系

**分解：** 将问题不断分解为较小的问题，直到每个最底层的问题都足够简单为止

---

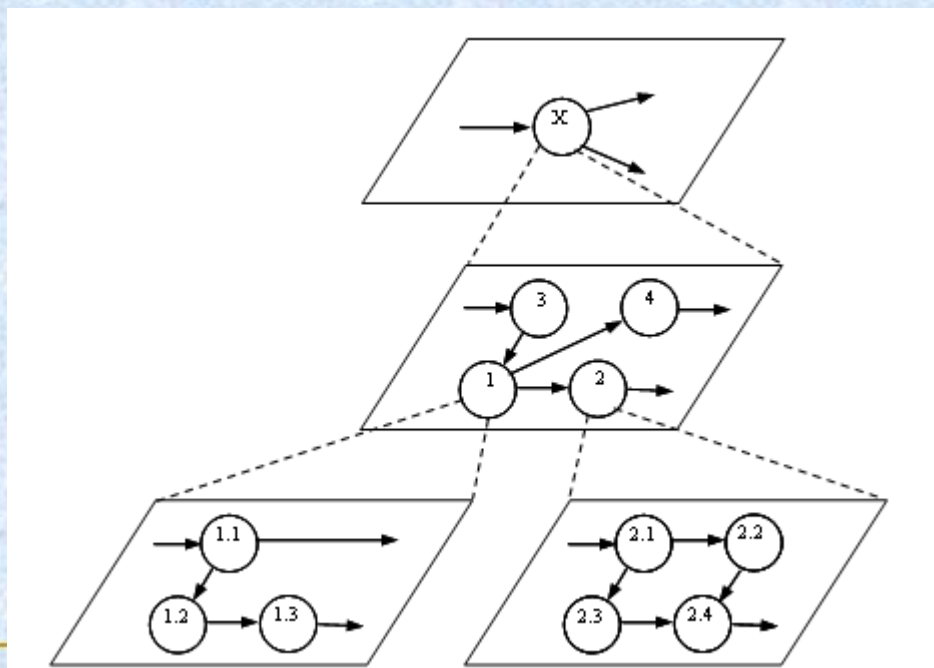




**抽象：**从作为整体的软件系统开始(第一层), 每一抽象层次上只关注于系统的输入输出

**分解：**将系统不断分解为子系统、模块……

随着分解层次的增加, 抽象的级别越来越低, 也越接近问题的解(算法和数据结构)





### 三、 结构化分析模型

- **数据字典**是模型的**核心**，包含软件使用和产生所有数据的描述。
  - **DFD数据流图**：用于功能建模，描述系统的输入数据流如何经过一系列的加工变换逐步变换成系统的输出数据流。
  - **ER实体关系图**：<sup>静态模型</sup>用于数据建模，描述数据字典中数据之间的关系。
  - **STD状态迁移图**：用于行为建模，描述系统接收哪些外部事件，以及在外部事件的作用下的状态迁移情况。
-





- **数据对象描述**

表示实体-关系图中每个数据对象的属性

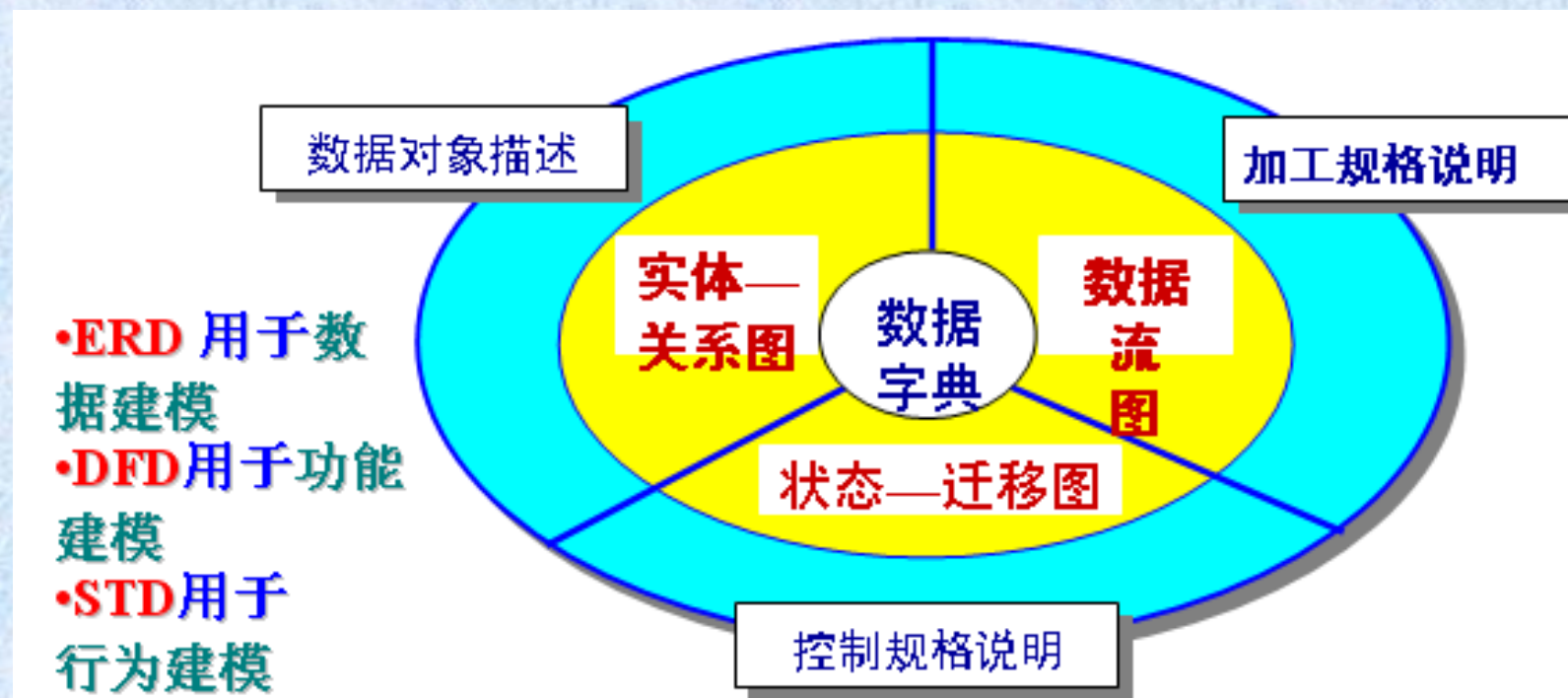
- **加工规格说明**PSPEC

描述数据流图的每个功能。

- **控制规格说明**CSPEC

描述软件控制的附加信息

---







## 四、 数据建模

- 数据对象、属性和关系

- 实体--关系图

实体—关系图是数据模型的基础，它描述数据对象、属性、及其关系。



# 1、数据对象、属性与关系

- 数据对象
  - 数据属性
  - 数据关系
-





## 数据对象

现实世界具有不同特征和属性的实体或事务的标识，计算机软件描述并处理的一组信息。如，事件、行为、角色、组织、地点、结构等。

数据对象只封装数据，包括：数据流、数据源、外部实体的数据部分，不封装操作。数据对象是相互关联的。

---



## 数据属性

用“标识符、符号串和值”标识，描述数据对象的性质。  
包括：

- (1) 命名            标识数据对象
- (2) 描述            描述数据对象的性质
- (3) 引用            建立数据对象之间的联系

说明：

- (1) 数据对象的属性是原子数据项，不包含内部数据结构。
  - (2) 数据对象的任何属性有且仅有一个属性值。
  - (3) 现实世界的实体具有许多属性，分析人员只能考虑与应用问题有关的属性。
-





## 数据关系

- 数据对象按照某种关系相互连接
  - 用对象-关系描述数据对象
  - 关系的命名及内涵应反映描述的问题
  - 删除与问题无关的关系
-



## 数据对象、属性与关系

例：汽车销售问题的数据对象、属性与关系

数据对象属性

制造商

购车用户

数据对象

汽车

汽车

关系

生产

购买





## 2 实体-关系图

描述系统所有数据对象的组成和属性，描述数据对象之间关系的图形语言。

- “一对一” (1: 1)

一个对象A关联一个对象B, 反之，一个对象B关联一个对象A。如，夫妻。

- “一对多” (1: N)

一个对象A关联多个对象B, 反之，一个对象B关联一个对象A。如，父子。

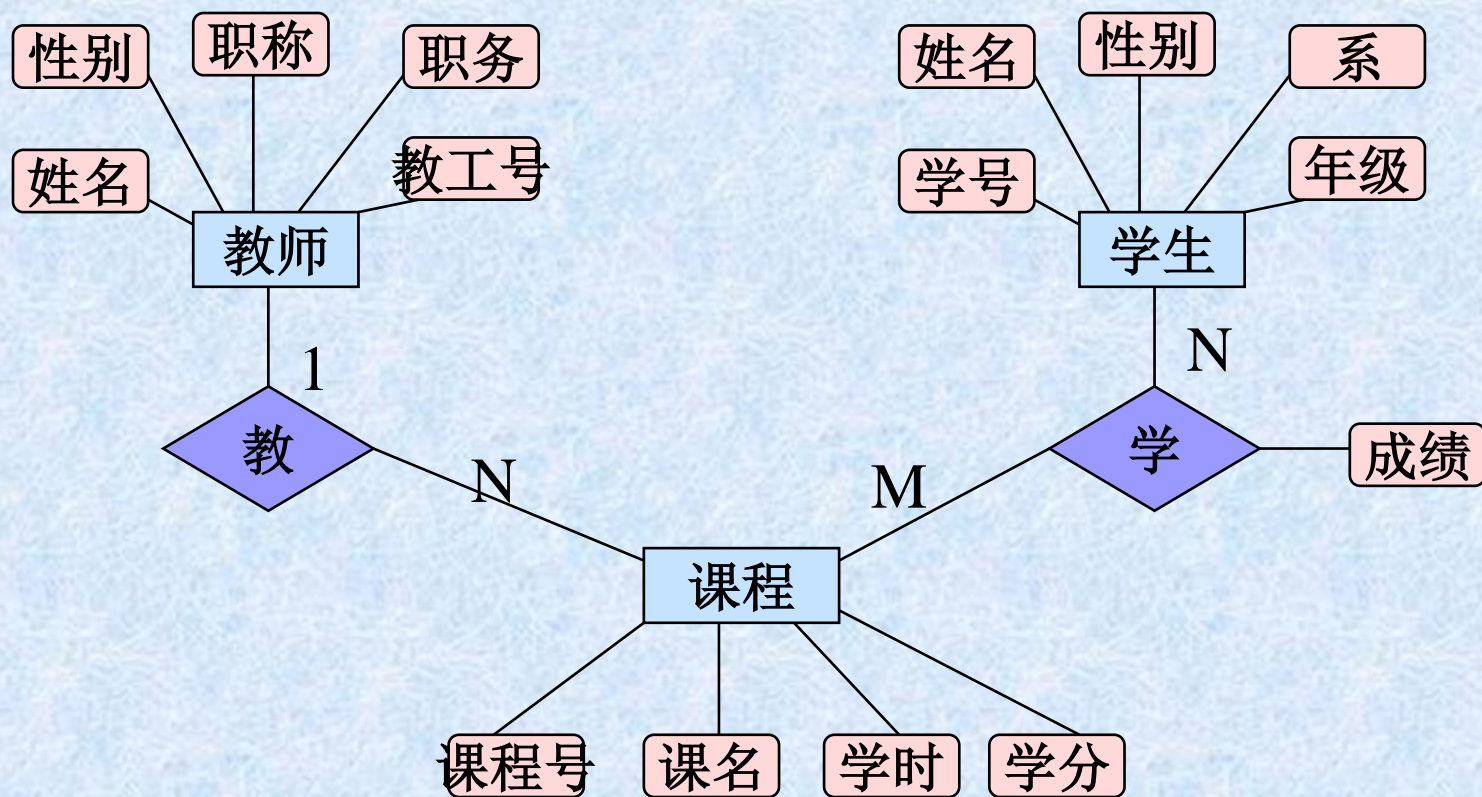
- “多对多” (N: M)

一个对象A关联多个对象B, 反之，一个对象B关联多个对象A。如，叔侄。

---



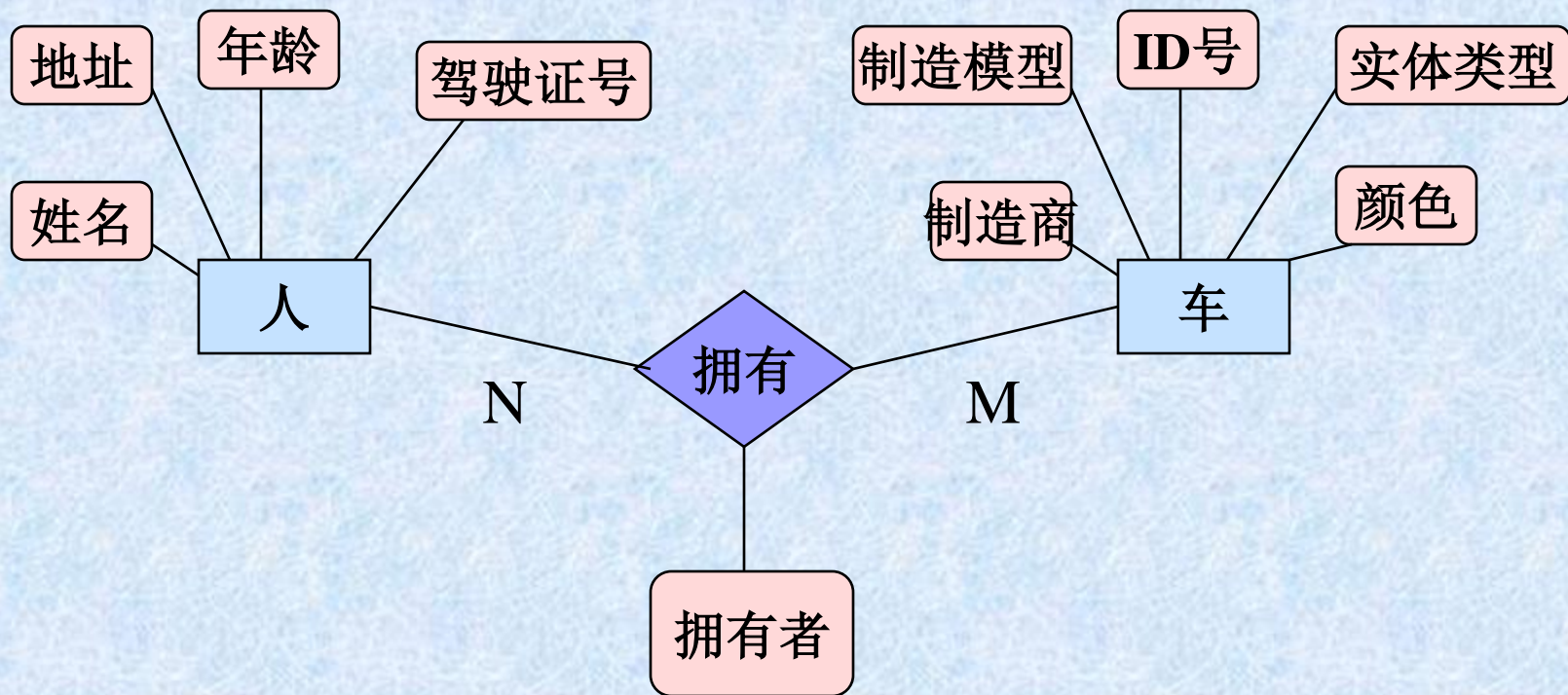
## 教师-学生-课程E-R 图







## 人与车关系E-R 图





## 五、 功能建模

用抽象模型的概念，按照软件内部数据传递、变换的关系，自顶向下逐层分解，直到找到满足功能要求的所有可实现的软件为止。

- 数据流图

- 数据字典





## 1 数据流图

- 基于计算机的信息处理系统由数据流和一系列的加工构成，这些加工将输入数据流加工为输出数据流
  - 数据流图描述数据流和加工
  - 数据流图用图形符号表示数据流、加工、数据源及外部实体
  - 数据流图具有层次结构，支持问题分解、逐步求精的分析方法
  - 它是数据驱动的数据流图，既可以表示基于计算机的系统，又可以表示软件
-



## 数据流图标记

外部实体

位于软件系统边界之外的信息生产者或消费者

转换

转换数据流的处理过程，又称泡(bubble)

数据流

在转换之间有向流动的数据项或数据项集合

数据源

为一个或多个转换提供数据源或数据存储服务的缓冲区、文件或数据库



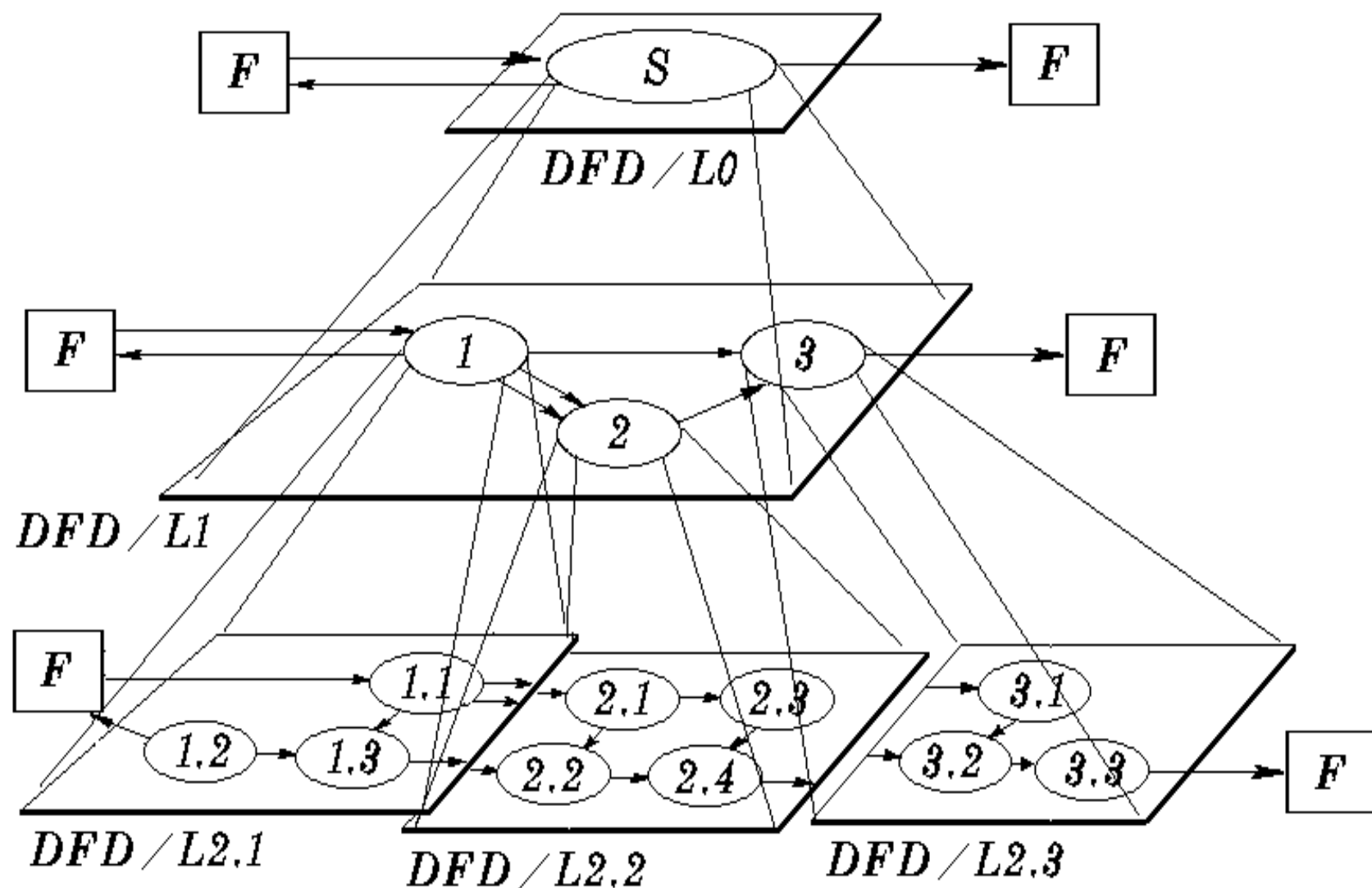


## 数据流图层次结构 DFD图精化

为了表达数据处理过程的数据加工情况，需要采用层次结构的数据流图。按照系统的层次结构进行逐步分解，并以分层的数据流图反映这种结构关系，能清楚地表达和容易理解整个系统

一个泡，关注输入和输出

- 在多层数据流图中，顶层流图仅包含一个加工，它代表被开发系统。它的输入流是该系统的输入数据，输出流是系统所输出数据
  - 底层流图是指其加工不需再做分解的数据流图，它处在最底层
  - 中间层流图则表示对其上层父图的细化。它的每一加工可能继续细化，形成子图。
-







## 顶层数据流图

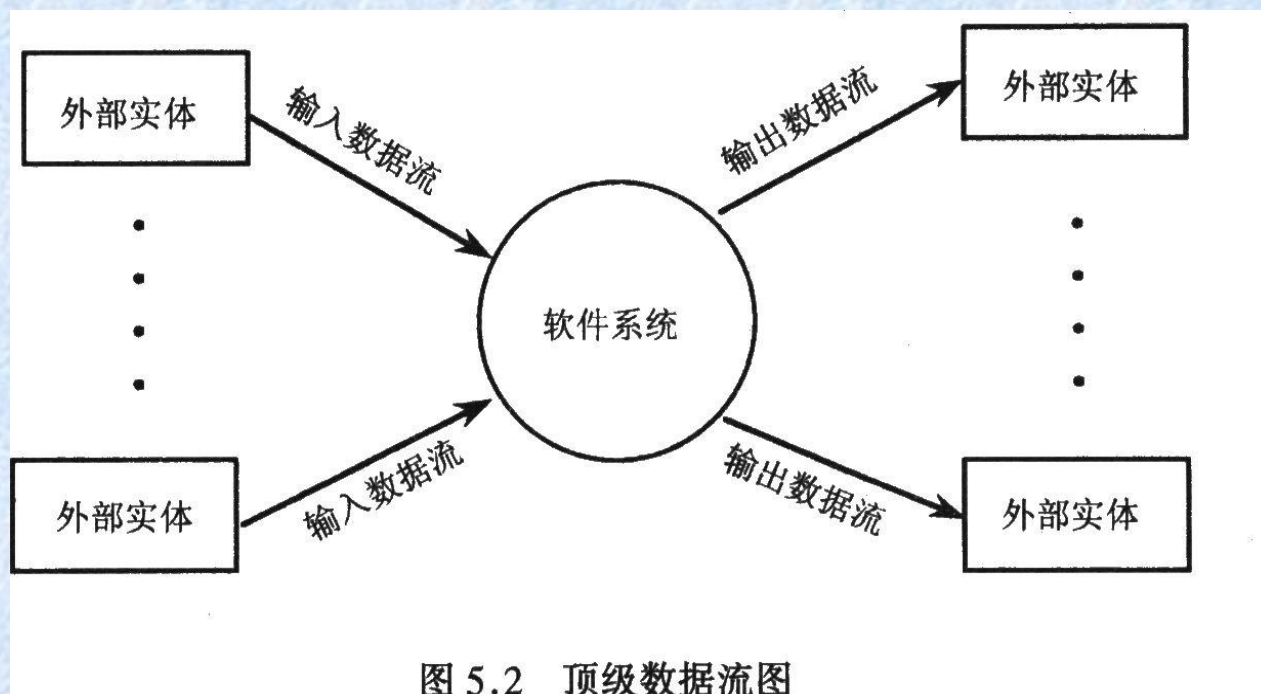


图 5.2 顶级数据流图

随着需求分析活动的深入，较高抽象级别的复杂加工逐步精化为一系列相互关联的数据流和子加工。



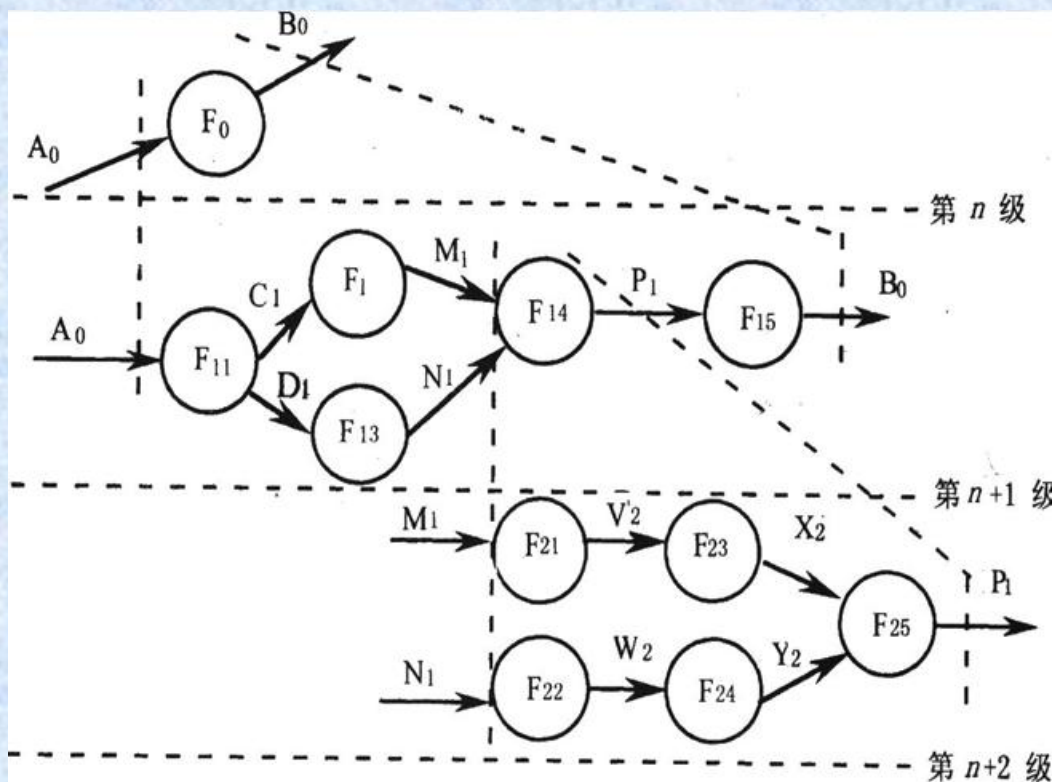
## 数据流图的精化与平衡

△逐层精化必须保持数据流图的平衡

△数据流与加工精化必须保持一致

图中A0输入B0输出，分解后保持一致

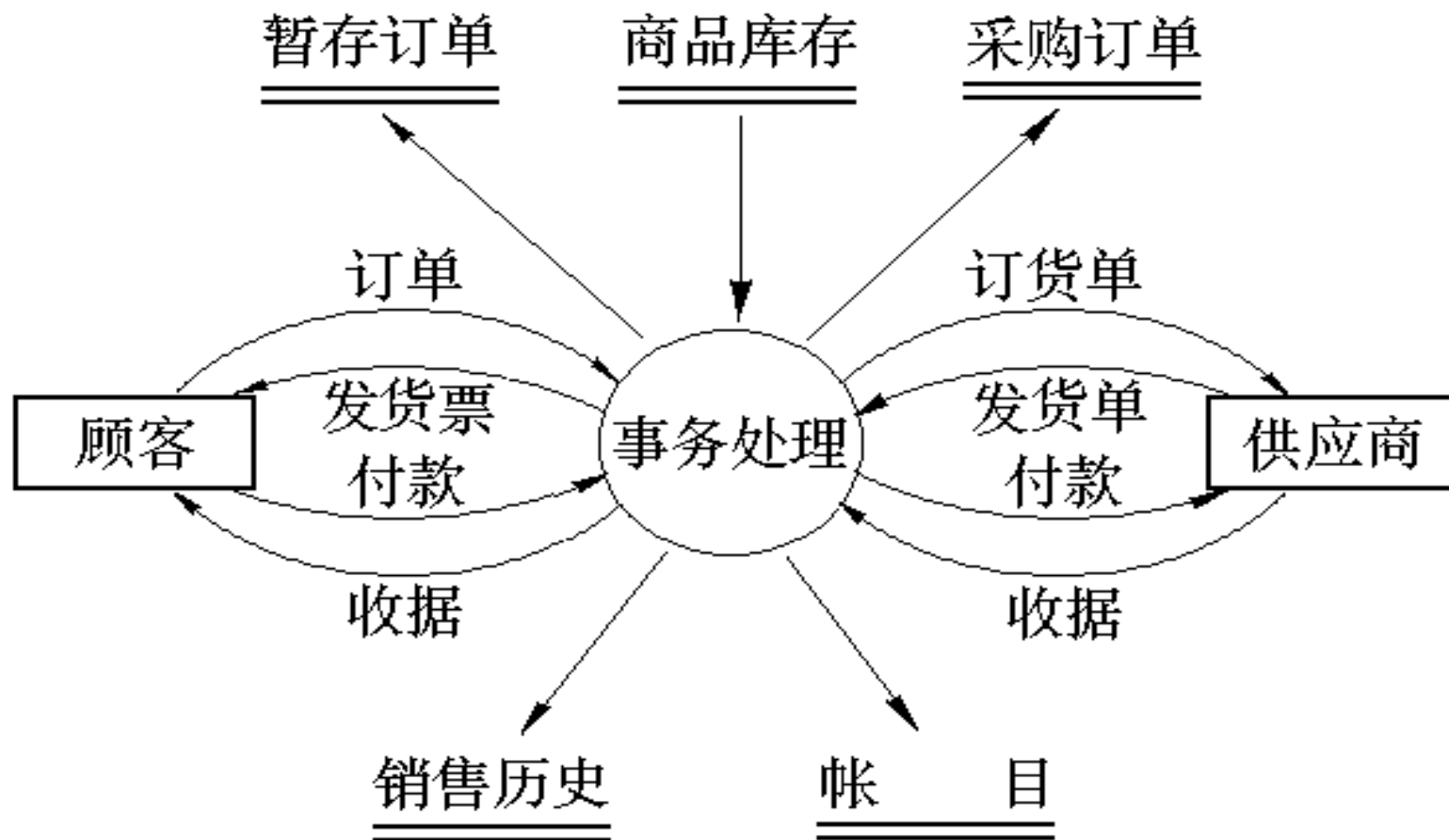
△需求分析活动只求对问题全面、清晰的理解，不考虑软件设计细节





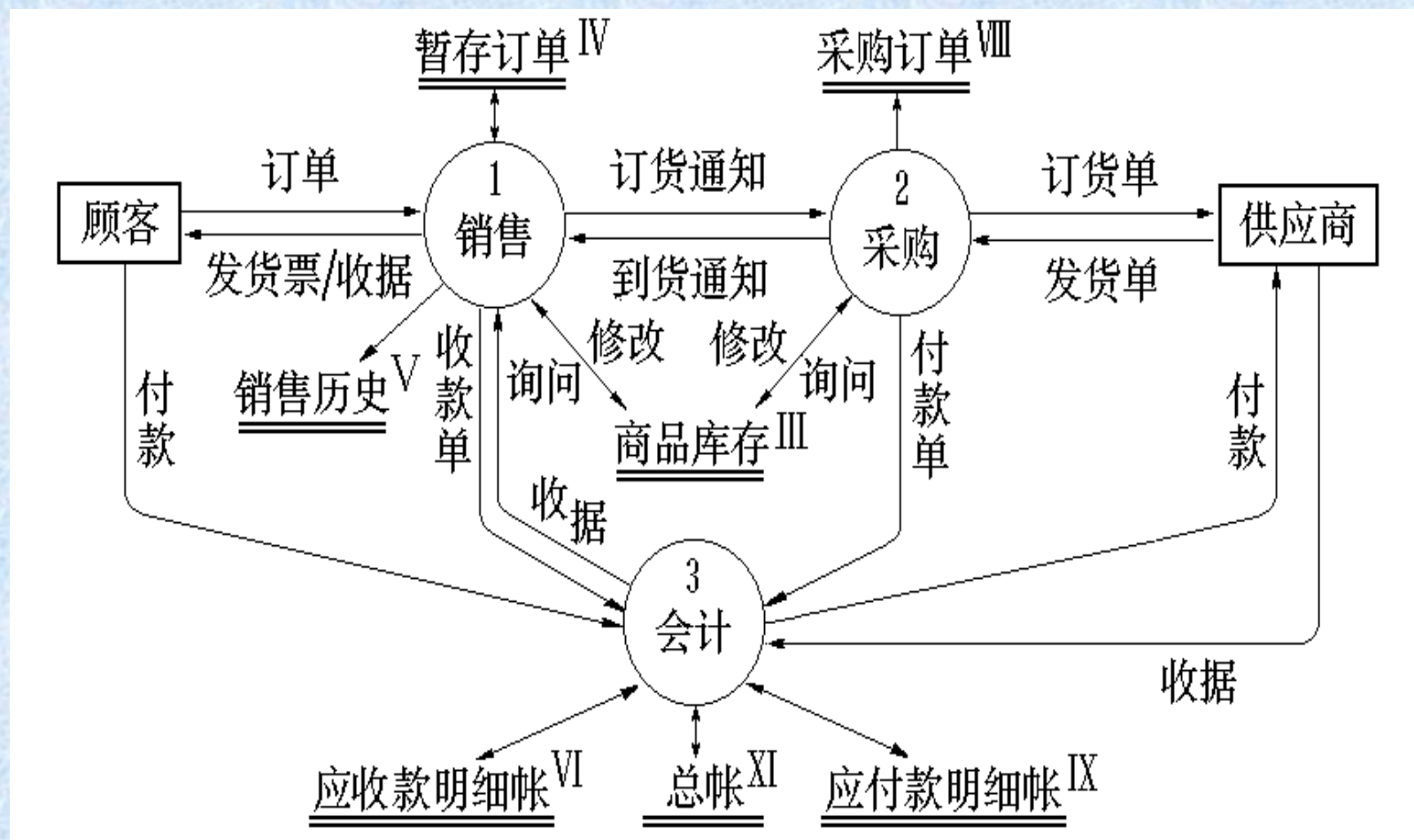


## 例子:商店业务处理系统





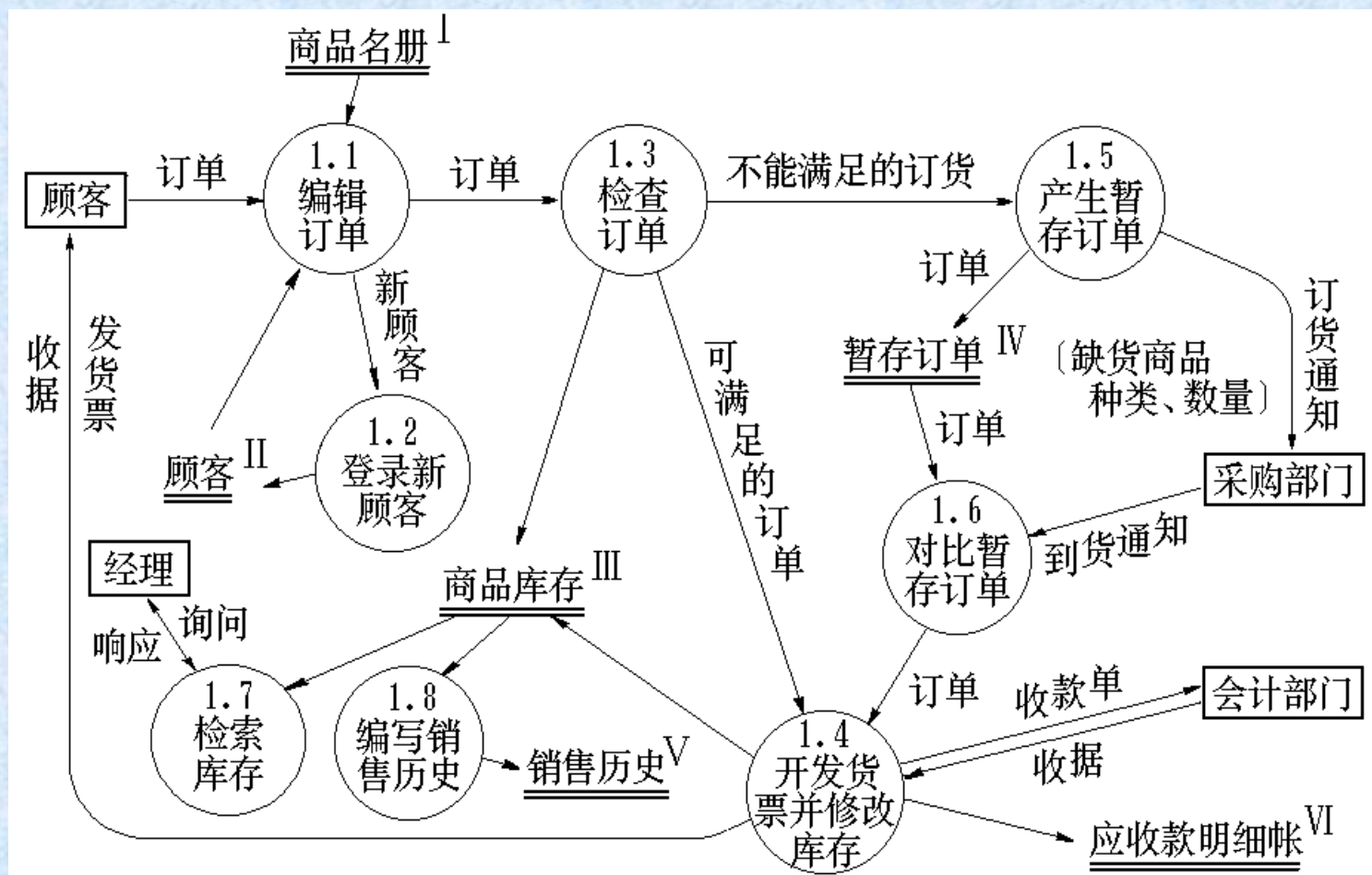
## 顶层数据流图







# 销售细化





## 2、数据字典

数据字典由数据条目组成，数据字典描述、组织和管理数据流图的数据流、加工、数据源及外部实体。

- 数据流图与数据字典是密不可分的，两者结合起来构成软件的逻辑模型
  - 数据字典由字典条目组成，每个条目描述DFD中的一个元素
  - 数据字典条目包括：数据流、文件、数据项(组成数据流和文件的数据)、加工、源或宿。
-





## 数据流条目的描述内容

- 名称：数据流名(可以是中文名或英文名)
  - 别名：名称的另一个名字
  - 简述：对数据流的简单说明
  - **数据流组成**：描述数据流由哪些数据项组成
  - 数据流来源：描述数据流从哪个加工或源流出
  - 数据流去向：描述数据流流入哪个加工或宿
  - 数据量：系统中该数据流的总量
    - 或者单位时间处理的数据流数量，如80000张/天
  - 峰值：某时段处理的最大数量
    - 如每天上午9：00至11：00处理60000张表单
  - 注解：对该数据流的其它补充说明
-





## 数据流组成

- 数据流组成是数据流条目的核心，它列出组成该数据流的各数据项，例如：
    - 培训报名单 = 姓名 + 单位 + 课程
    - 运动员报名单 = 队名 + 姓名 + 性别 + { 参赛项目 }
  - 当一个数据流的组成比较复杂时，可以将其分解成几个数据流，例如：
    - 课程 = 课程名 + 任课教师 + 教材 + 时间地点
    - 时间地点 = { 星期几 + 第几节 + 教室 }
-





## 文件条目的描述内容

- 名称：文件名
  - 别名：名称的另一个名字
  - 简述：对文件的简单说明
  - 文件组成：描述文件的记录由哪些数据项组成(与数据流条目中的文件组成描述方法相同)
  - 写文件的加工：描述哪些加工写文件
  - 读文件的加工：描述哪些加工读文件
  - 文件组织：描述文件的存储方式(顺序、索引)，排序的关键字
  - 使用权限：描述各类用户对文件读、写、修改的使用权限
  - 数据量：文件的最大记录个数
  - 存取频率：描述对该文件的读写频率
  - 注解：对该文件的其它补充说明
-





## 数据项条目的描述内容

- 名称：数据项名
  - 别名：名称的另一个名字
  - 简述：对数据项的简单描述
  - 数据类型：描述数据项的类型，如整型、实型、字符串等
  - 计量单位：指明数据项值的计量单位，如公斤、吨等
  - 取值范围：描述数据项允许的值域，如1...100
  - 编辑方式：描述该数据项外部表示的编辑方式，如23,345.67
  - 与其它数据项的关系：描述该数据项与数据字典中其它数据项的关系
  - 注解：对数据项的其它补充说明
-





## 加工条目的描述内容

- 名称：加工名
  - 别名：名称的另一个名字
  - 加工号：加工在DFD中的编号
  - 简述：对加工的功能的简要说明
  - 输入数据流：描述加工的输入数据流，包括读哪些文件名
  - 输出数据流：描述加工的输出数据流，包括写哪些文件名
  - 加工逻辑：简要描述加工逻辑，或者对加工规约的索引
  - 异常处理：描述加工过程中可能出现的异常情况，及处理方式
  - 加工激发条件：描述执行加工的条件，如，“身份认证正确”，“收到报名单”
  - 执行频率：描述加工的执行频率，如，每月执行一次
  - 注解：对加工的其它补充说明
-



## 源或宿条目的描述内容

- 名称：源或宿的名（外部实体名）数据从源流出，流入宿
  - 别名：名称的另一个名字
  - 简要描述：对源或宿的简要描述（包括指明该外部实体在DFD中是用作“源”，还是“宿”，还是“既是源又是宿”）
  - 输入数据流：描述源向系统提供哪些输入数据流
  - 输出数据流：描述系统向宿提供哪些输出数据流
  - 注解：对源或宿的其它补充说明
-





## 数据字典的描述符号

符 号	名 称	举 例
=	定义为	$x = \dots$ 表示 $x$ 由 $\dots$ 组成
+	与	$a + b$ 表示 $a$ 和 $b$
[..., ...]	或	$[a, b]$ 表示 $a$ 或 $b$
[...   ...]	或	$[a   b]$ 表示 $a$ 或 $b$
{...}	重复	$\{a\}$ 表示 $a$ 重复0或多次
$\{...\}_m^n$	重复	$\{a\}_3^8$ 表示 $a$ 重复3到8次
(...)	可选	$(a)$ 表示 $a$ 重复0或1次
"..."	基本数据元素	" $a$ " 表 $a$ 是基本数据



## 六、行为建模

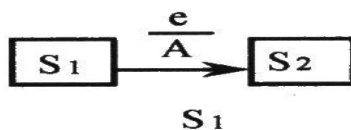
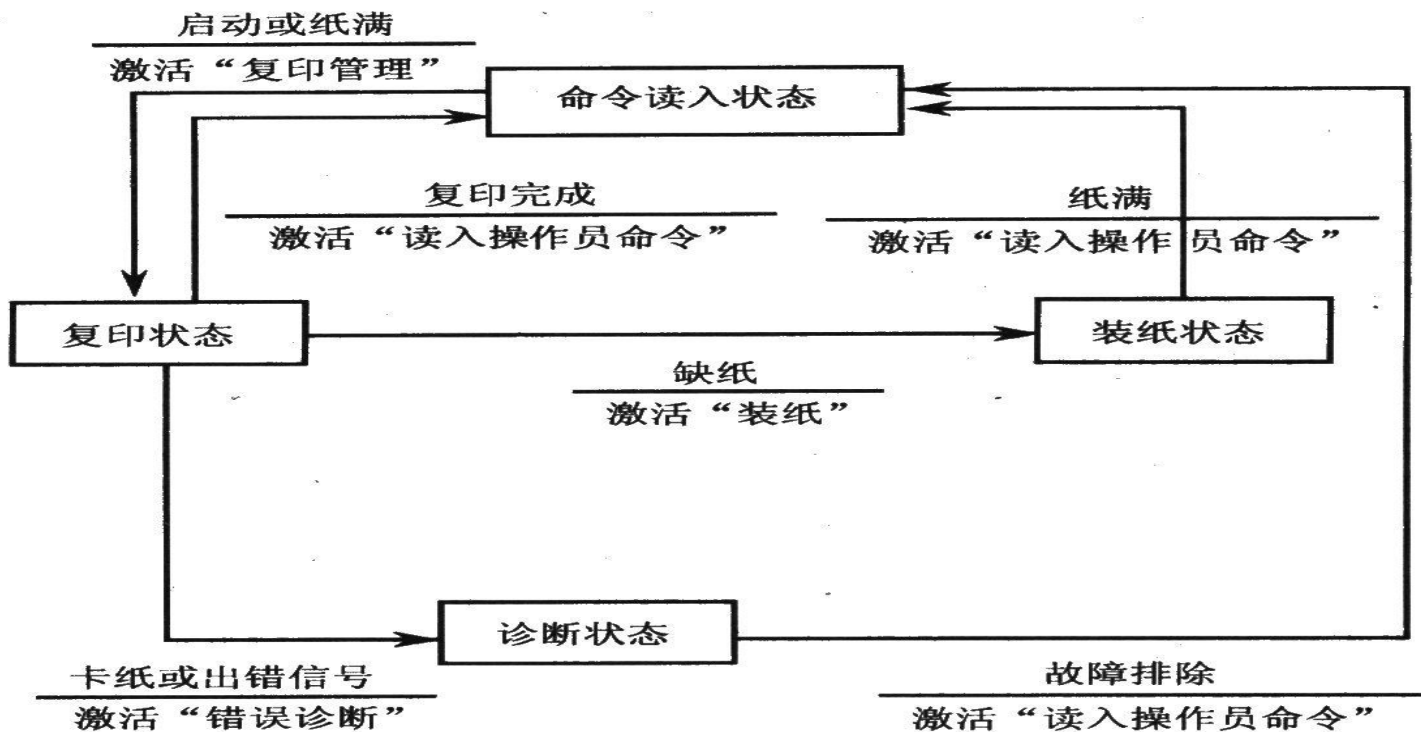
状态变迁图通过描述状态以及导致系统改变状态的事件来表示系统的行为。每个状态代表系统的一种行为模式

---





## 状态变迁图



表示：事件e使系统由状态 $S_1$ 变为 $S_2$ ，并引发行为A

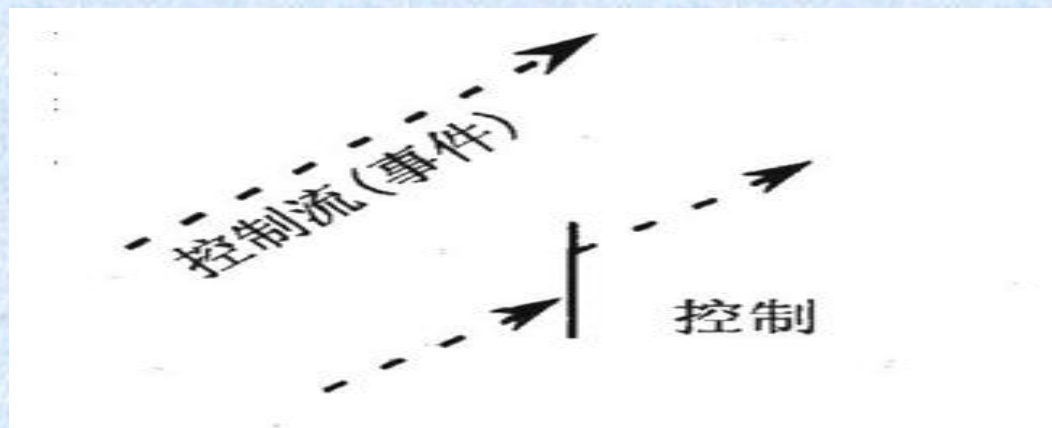
(c) 状态转换图



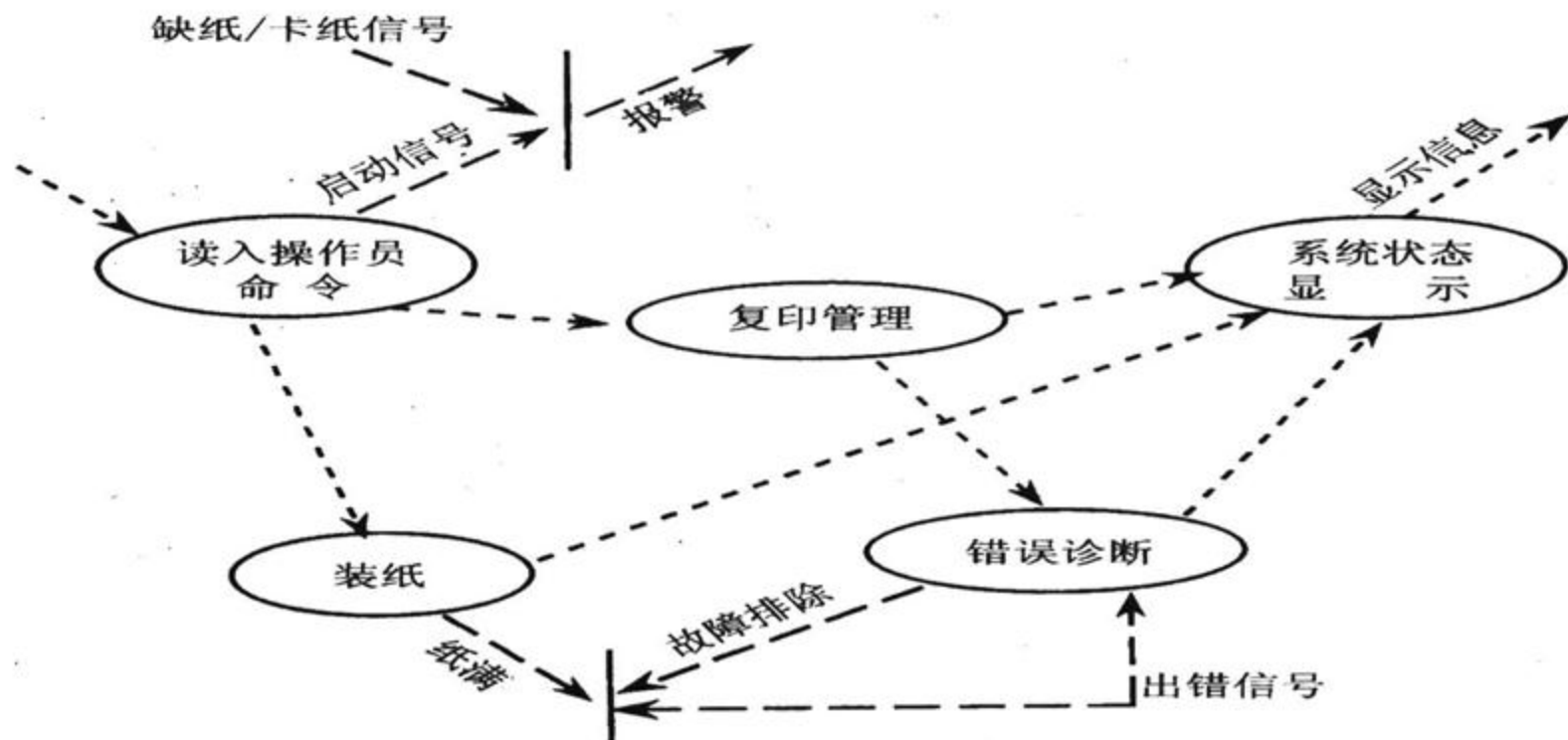
## 七、控制流建模（数据流图的扩充）

- 实时嵌入式系统存在事件驱动机制，通过系统的加工产生控制信息。
  - 系统对事件信息的加工必须在指定的时间内完成。
  - 实时嵌入式系统的描述不仅需要数据流图，而且还需要控制流图CFD。
-





- 引进两个新的图形记号，用虚线表示控制流或事件流；
- 虚符号和实符号分开，专门定义控制流图CFD；
- CFD定义控制流、表示控制加工並引入粗短线表示控制规格说明；  
控制规格说明指明
  - (1) 当事件发生或控制信号被感知时，软件的行为
  - (2) 当某一事件发生时，激活相关加工部件



图元 表示省略了名称的数据流， 则表示控制流（事件）





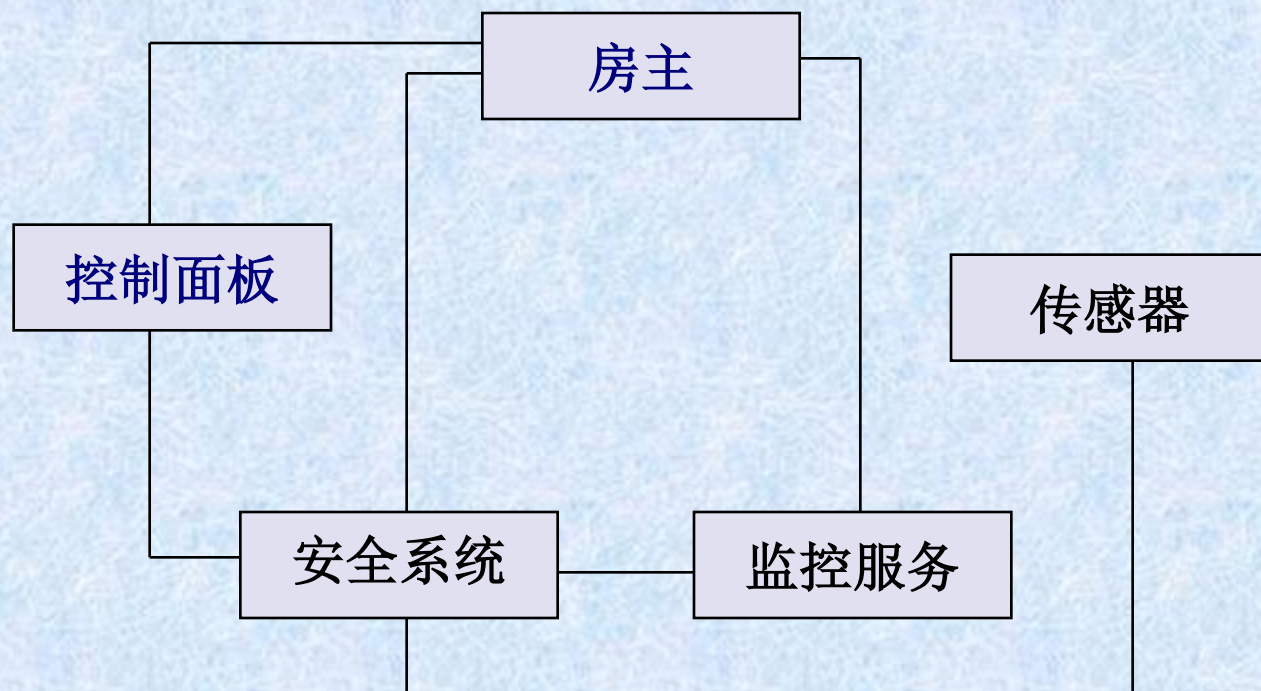
## 八、 实例分析---Safe Home

- 实体-关系图
  - 创建数据流模型
  - 过程规格说明
  - 控制流模型
-



## 实体-关系图

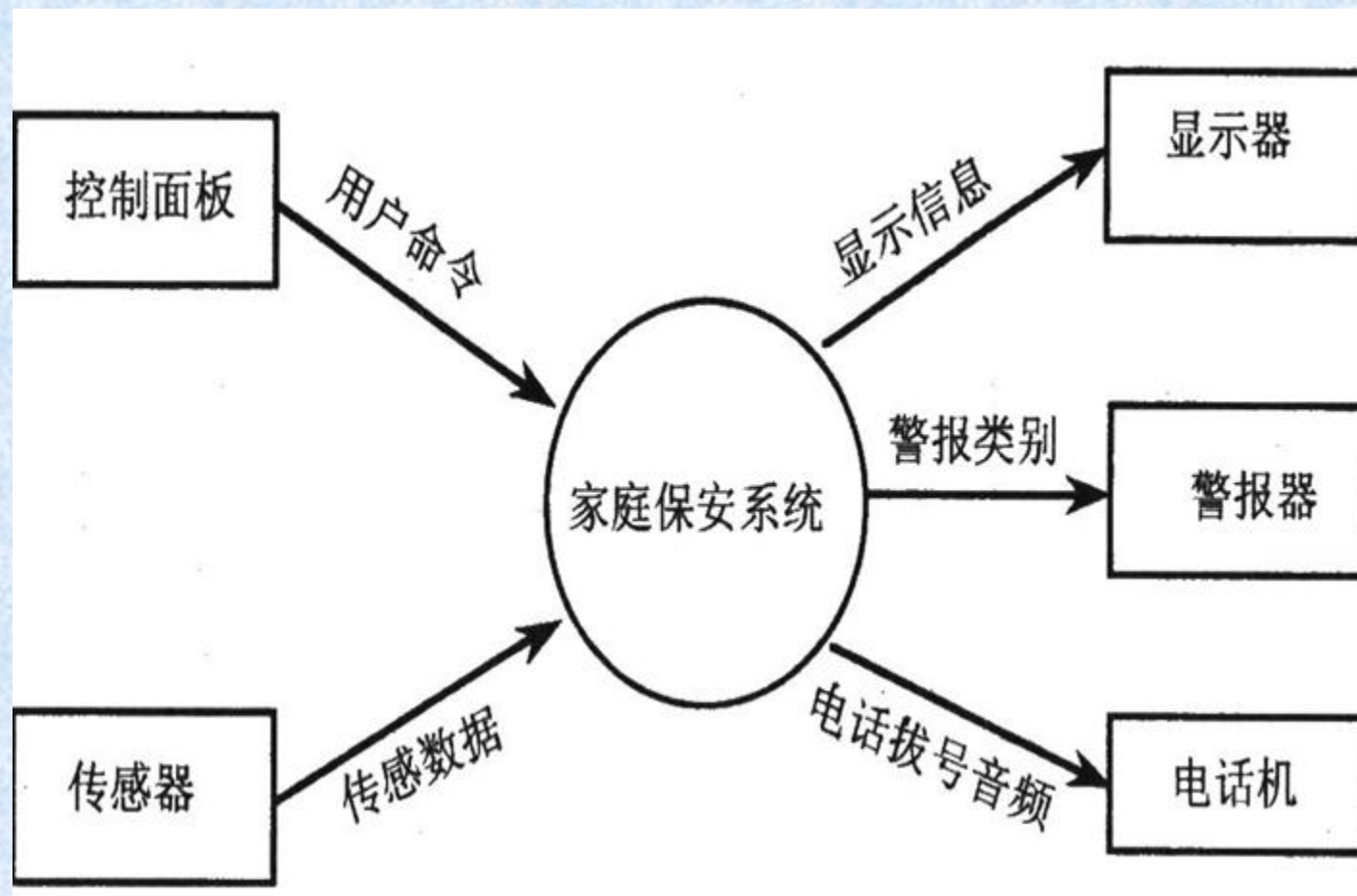
- 房主
- 控制面板
- 传感器
- 安全系统
- 监控服务



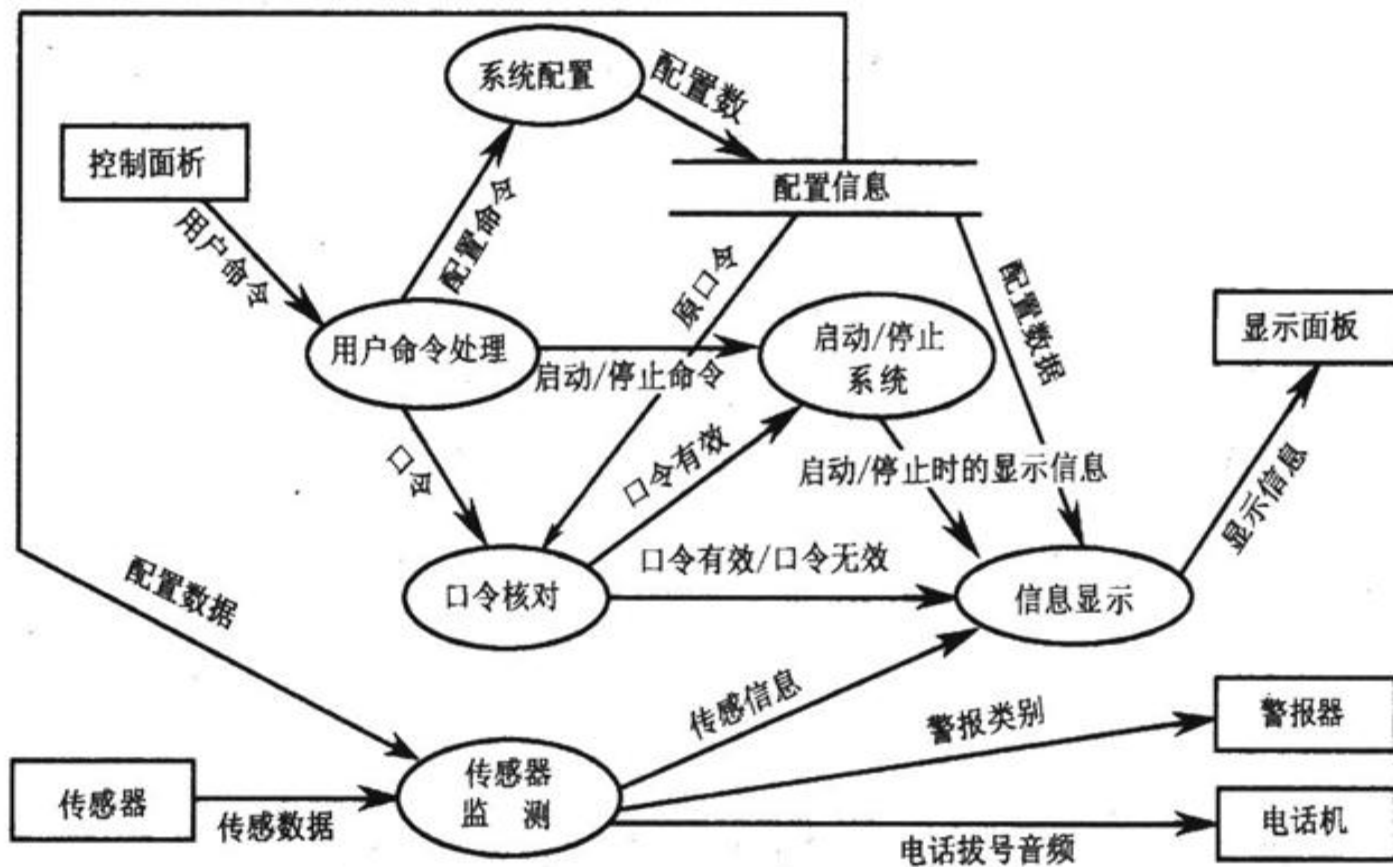




## 数据流模型

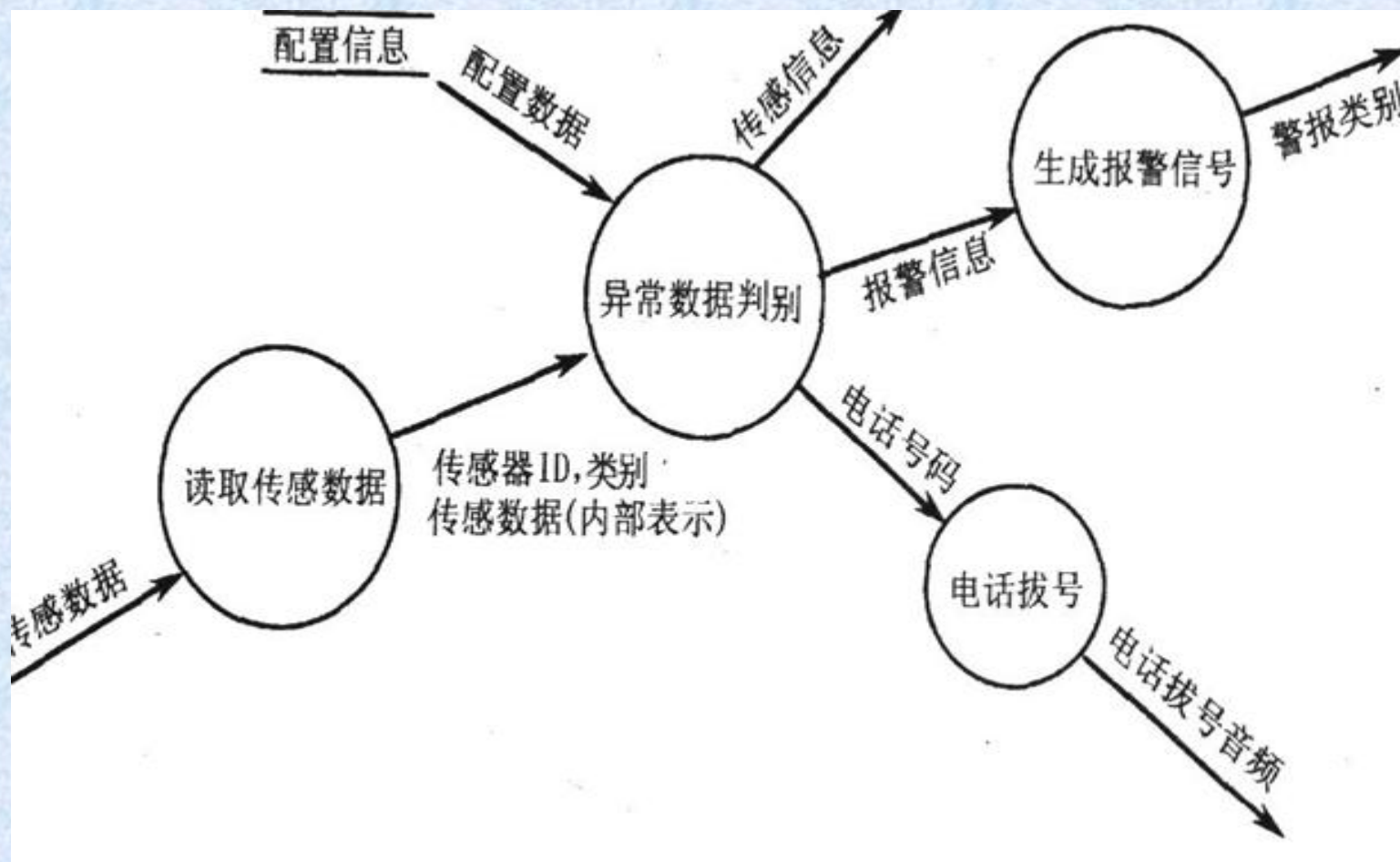


顶（0）级数据流图



1级数据流图

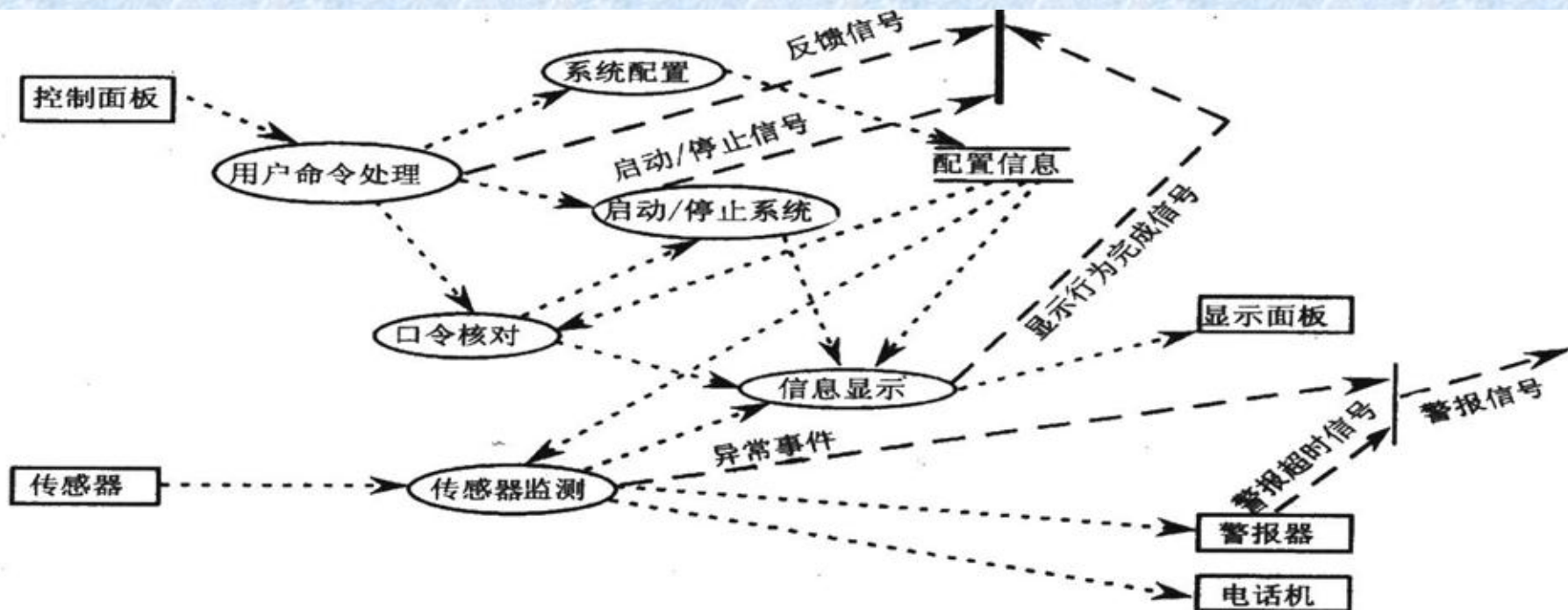




## 2级数据流图-对传感器检测的分解



## 控制流模型

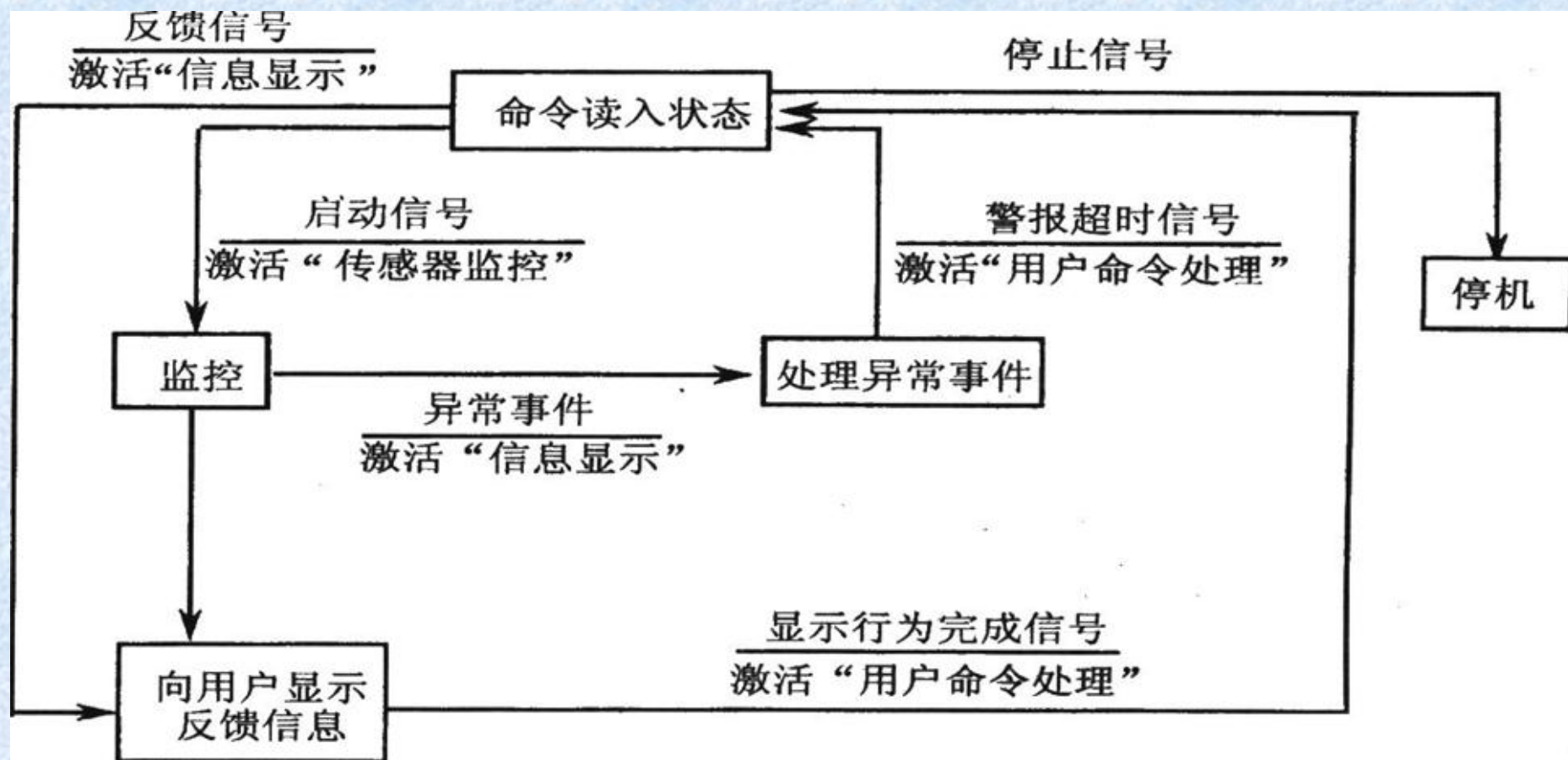


- 异常事件：当“传感器监测”发现异常情况后发送该事件。
- 反馈信号：“用户命令处理”在接收用户命令后发送该事件，以便显示命令处理状况。
- 警报超时信号：如果警报器拉响后在指定时间延迟范围内用户仍未处理当前警报，那么警报器发送该事件。
- 显示行为完成信号：一旦显示行为完成。“信息显示”发送完成信号。





## 行为模型





## 过程规格说明

- 对数据流图无须再分解的处理功能，可借助结构化的自然语言、“程序设计的语言（PDL）”、数学方程、表、图或图表等工具进行描述。
  - 必须为数据流模型的每个加工园盘提供相应的加工过程规格说明。
-