# 模型和方法：结构化需求分析

# 结构化需求分析的由来

* 1. 目的：
     1. 用一种图形符号体系来表示数据、表示对数据变换的处理，这些处理最终能被映射到软件体系结构的设计中

# 结构化分析模型的组成

* 1. 分析模型的主要目标
     1. 描述客户的需求；
     2. 建立创建软件设计的基础；
     3. 定义在软件完成后可以被确认的一组需求。

和面向对象需求分析方法的目的基本一致

* 1. 模型组成



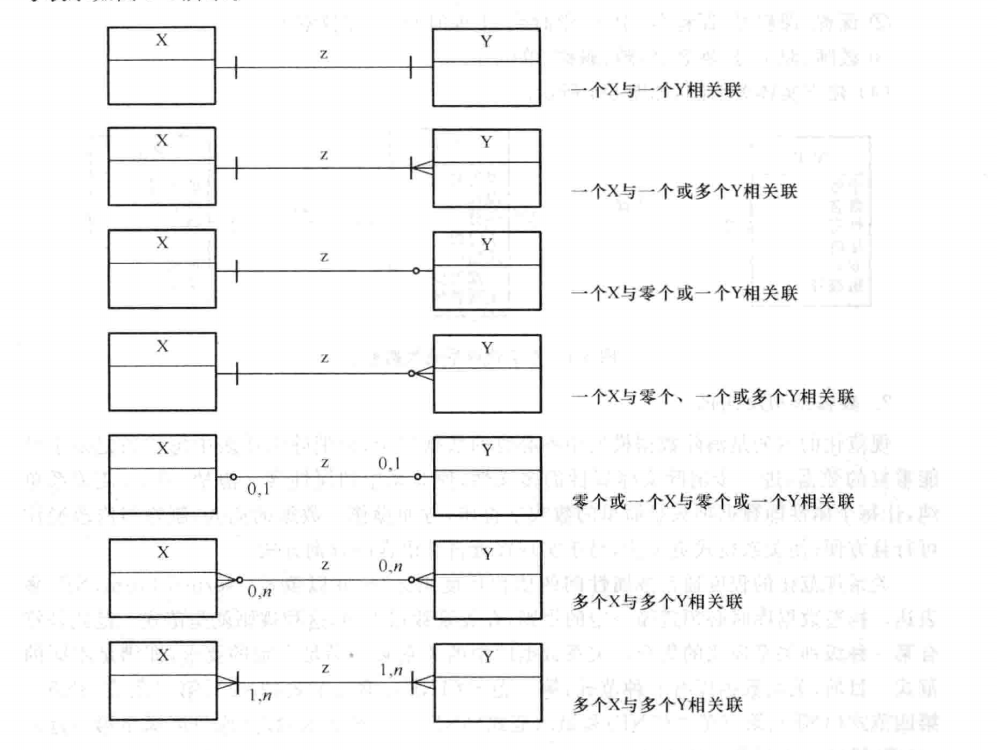
“数据驱动”思想，核心是数据词典，

视图：实体关系图（数据建模）、数据流图（功能建模、数据流）、状态迁移图（行为建模）

文档：数据对象描述、加工规格说明、控制规格说明

# 三、数据建模

* + 1. 软件系统本质是信息处理系统，在开发过程中必须考虑数据和数据处理两个方面
    2. 概念性数据模型是一种面向问题的数据模型。从用户角度看到的数据
    3. 表示方法：ER法（实体-关系法），也就是ER模型
    4. 作用：
       1. 确定系统处理的若干数据对象
       2. 确定数据对象的组成
       3. 数据对象的属性描述——和前一章的概念类描述类似
       4. 数据对象位于何处
       5. 数据对象和其他数据对象的关系
       6. 数据对象和变换对象的处理之间的关系——用户对于数据对象的处理要求
    5. **数据对象（实体）**
       1. 与目标系统相关的所有实体成为数据对象，它描述了数据对象的名称和属性，通常将数据对象称为实体。
       2. 外部实体（产生或者使用信息的事物）、事物、事件、角色、组织单位、地点、结构
       3. 例如：对于北斗系统来说，使用该系统的手机属于“外部系统”
    6. **数据对象的属性和关系**
       1. 属性：数据对象**区别于其他数据对象**的特征和性质
       2. 属性的用处：
          1. 数据对象命名
          2. 描述数据对象实例
          3. 建立与其他数据对象之间的联系
       3. 关系：数据对象之间的特定连接，由问题的语境定义，关系并不是一成不变的【工人（生产/使用）工具】。
    7. **数据对象的基数**
       1. 数据建模的基本元素
          1. 数据对象
          2. 属性
          3. 关系
       2. 基数：描述系统的数据对象之间的出现次数有无必然联系，称为实体-关系对的基数。
       3. 基于基数，实体之间的关联有如下分类：
          1. 一对一：人->身份证
          2. 一对多：父母->孩子
          3. 多对多：学生->老师【多:意味着0~n】
    8. **实体关系图**
       1. 图形化表示实体之间的关系
       2. 基本构件：实体【实体名字+实体属性构成的矩形框】、属性、关系



* + 1. **数据结构规范化**
       1. 基数关系在关系型数据库会造成数据存储的冗余，为了消除冗余，需要规范化。——数据范式
       2. 使用规范化的好处：
          1. 消除多义性：属性含义清晰
          2. 使得关系简单化，方便操作
          3. 关系模式更加灵活：易于实现近乎自然语言查询
       3. 范式级别高：
          1. 冗余程度随着范式提高而变小【第一范式冗余度最大】
          2. 存储自身的过程越复杂【数据被分解成多张表】
          3. 数据的存储结构和问题域的结构之间的匹配程度随之下降，在需求发生变化的时候，数据稳定性下降。
          4. 访问的表增多，性能或者处理速度下降
       4. 一范式（1NF）：
          1. 所有属性都是“单纯域”，即不出现“表中有表”；表的每一行只包含一个实例的信息。
       5. 二范式（2NF）：
          1. 非主属性**完全函数依赖**于关键字：完全函数依赖表示，非主键元素，必须完全依赖主键组合的全部成员，而不是仅仅依赖于主键组合中的一个子集。
          2. 首先必须**满足1NF**，然后要求表中的每个实例或行必须可以被**唯一地区分。**
          3. 需要添加一个实例的唯一标识列，称为主关键字或**主键**、主码。
          4. 学生和课程两个数据对象多对多，如果放在一张表中，会产生数据异常、数据冗余、插入困难【不选课的学生，课程信息为空】、删除冗余【删除学生信息，会删除掉所有的课程信息】。
       6. 三范式（3NF）：
          1. 在满足第二范式的基础上，要求**非主属性相互独立**，即任何非主属性间不存在函数依赖。
          2. 可能导致数据冗余、删除操作时出现一些异常
          3. 将存在依赖的属性单独拎出来

# 四、功能建模

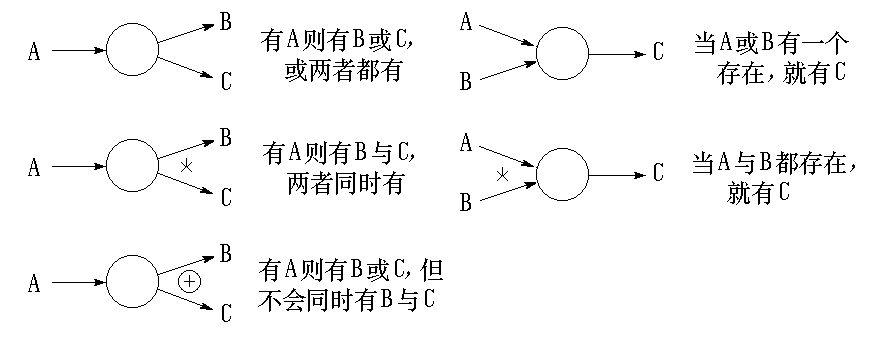
* + 1. **结构化分析方法，以数据出发；**
    2. **功能建模则着重于功能的表示**
    3. **数据流图**
       1. **可以提供功能建模，抽象表示系统或者软件，也可以提供数据流建模的机制，自顶向下**
       2. **又称为“泡泡图”**
       3. **基本元素：**
          1. 
          2. 加工一般用动词表示一定的功能，加工原料是由外部实体或者别的提供的数据；
          3. 分层级的概念，每一个层级的加工涵盖的范围和力度存在不同；

系统级加工【涵盖面广】

原子级加工【细粒度】

* + - * 1. 数据传输的终点，通常代表的是使用系统的角色，产生数据的外部实体常称为“数据源”，接受流出数据的外部实体常称为“数据潭”，
        2. 数据流：系统加工处理的对象、是最核心的组成部分；最初由外部实体产生，最终流向外部实体，数据流具有方向性；**以名词或者名词短语进行数据流表示；**

复杂模型内部会有控制数据流等细分的数据流类型；

* + - * 1. **数据存储：保存数据，可以是表、文件、临时数组、临时队列等数据存储形式，是有方向性的，存在输入和输出，**
      1. **数据流和加工的关系：**
         1. 
         2. 约束：\*表示两个或者多个分支皆出现；⊕表示分支中只出现一个；
      2. **如何绘制数据流图**
         1. **分层**

**顶层数据流图：顶层流图仅包含一个加工，它代表被开发系统，其作用在于表明被开发系统的范围，以及它和周围环境的数据交换关系。 【明确问题域和系统的功能范围】**

**中间层数据流图：表示对其上层父图的细化。它的每一加工可以继续细化，形成子图。中间层次的多少视系统的复杂程度而定。 【功能细化】**

**底层数据流图：是指加工不须再做分解的数据流图，称为“原子加工”。 【】**

* + - * 1. **表示方式**

**分层，注意命名**

* + - * 1. **评价标准（要求）**

**数据平衡——对数据流图的评价准则**

* + - * 1. **绘制步骤**

**先找系统的数据源点与汇点。它们是外部实体，由它们确定系统与外界的接口。**

**找出外部实体的输出数据流与输入数据流。**

**在图的边上画出系统的外部实体。**

**从外部实体的输出数据流（即系统的源点）出发，按照系统的逻辑需要，逐步画出一系列逻辑加工，直到找到外部实体所需的输入数据流(即系统的汇点)，形成数据流的封闭。**

**按照下面所给的原则进行检查和修改。**

**按照上述步骤，再从各加工出发，画出所需的子图。**

* + - 1. **实时系统的数据流图**
         1. **实时系统的要求：**

**在时间连续的基础上接收或产生数据流；**

**贯穿系统的控制信息和相关的控制处理；**

**多任务的情况下可能会遇到同一个加工的多个实例；**

**系统状态以及导致系统状态迁移的机制。**

* + - * 1. **引入元素**



连续数据流：数据需要是连续不断的

**控制项或者事件：表示控制条件【？】**

**控制加工：对控制信息进行处理**

**控制存储：存储控制信息**

**加工的多个实例：处理并发等情况**

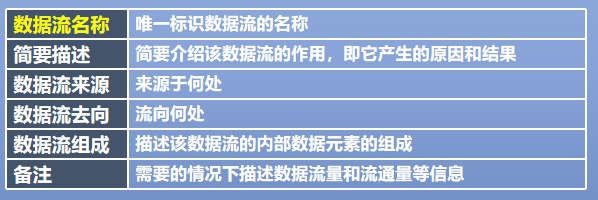
* + - * 1. **事件可以作为普通数据加工的输入数据流，控制加工也可以接收普通的输入数据流。**
      1. **数据流图例子：**

# 五、行为建模

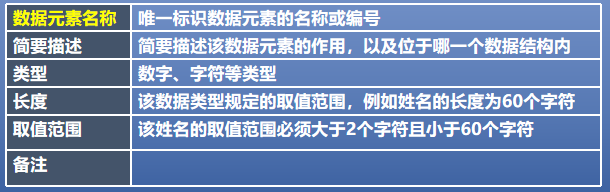
* + 1. 目的：直观分析系统的动作，从特定的视角出发描述系统的行为——动态分析
    2. 常用的结构化动态分析的方法：
       1. 状态迁移图
       2. 时序图
       3. **Petri图**
    3. 状态迁移图
       1. 描述状态以及导致状态改变的事件，从而描述系统的行为。
       2. 分层的状态迁移图：
    4. **Petri网**
       1. 目的：最初用来处理并发系统中的同步问题、资源竞争以及死锁问题
       2. 有向图，包含四个基本元素：一组位置P、一组转换T、输入函数I、输出函数O
       3. 标记/权标：
          1. 标记在位置中的出现表明了处理要求的到来。
          2. 当每个输入位置所拥有的标记数大于等于从该位置到转换的线数时，就允许转换。
          3. 标记向量：（……Pi的标记……）
          4. 在转换T的过程中，会存在权标的转移，具体的转移情况由转移的线数决定

# 六、数据词典

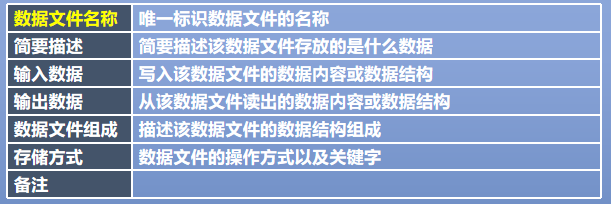
* 1. 用途：
     1. 对于**数据流图**中出现的所有被命名的图形元素加以定义，使得每一个图形元素的名字都有一个确切的解释
     2. 其定义应是严密的、精确的，不可有半点含混并消除二义性，
  2. 内容
     1. 数据流词条——数据流



* + 1. 数据元素词条——基本组成，原子级，认为成是构成其他部件的更小单位



* + 1. 数据文件词条——存储部件



* + 1. 加工词条——加工
       1. 须对加工的逻辑或规则进行描述，其方法有判定表、判定树或结构化英语
          1. 写基本加工逻辑的说明应满足要求：

数据流图**每一个基本加工都必须有一个加工逻辑说明；**

加工逻辑说明描述**基本加工如何把输入数据流变换为输出数据流的加工规则**（参考业务规则）；

加工逻辑说明必须**描述实现加工的策略**而不是实现加工的细节。

* + - 1. 结构化英语

介于自然语言和形式化语言之间的半形式化语言

使用有限的词汇和有限的语句来描述加工逻辑。

词汇表组成：

英语命令动词、

数据词典中定义的名字、

有限的自定义词

控制结构关键词：

IF\_THEN\_ELSE

WHILE\_DO

REPEAT\_UNTIL

CASE\_OF

* + - 1. 判定表

条件桩—左上部分：列出了各种可能的条件。

条件项—右上部分：给出各个条件的条件取值的组合。

动作桩—左下部分：列出了可能采取的动作。

动作项—右下部分：是和条件项紧密相关的，它指出了在条件项的各种取值的组合情况下应采取什么动作。

* + - 1. 判定树

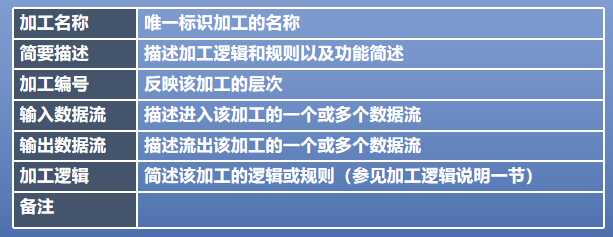
更加直观

* + - 1. **在表达一个基本加工逻辑时，结构化英语、判定表和判定树常常交叉使用，互相补充。**

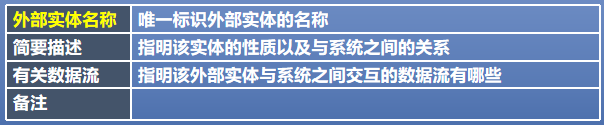
总之，**加工逻辑说明是结构化分析方法的一个组成部分，对每一个加工都要加以说明**。

应当**以结构化英语为主，对存在判断问题的加工逻辑，可辅之以判定表和判定树**。

* + - 1. 基本格式



* + 1. 外部实体——外部实体



1. **数据词典是面向对象的需求分许所没有的**
2. 需求分析（面向对象、结构化）均生成一个文档
3. 复习
   1. 需求分析的对象、任务、目标
   2. 领域模型是模型化过程（当前系统->物理模型）
   3. 需求分析的类型
   4. 领域建模绘制、用例建模绘制
   5. 面向对象的需求分析建模——P15
   6. 类的关系——组合和聚合、如何通过软件类判断是不是依赖、组合等关系
   7. 用例建模：组成、会画用例图、用例和子用例的关系（包含、拓展）、操作契约的前、后置条件、系统顺序图两个组成：角色、消息
   8. 用例模型的基本结构——P40
   9. 功能建模要会绘制
   10. 数据流图至少应该画到第二层

