

## 3.1 需求分析的任务

## 3.2 与用户沟通获取需求的方法

## 3.3 分析建模与规格说明

## 3.4 实体联系图（不讲）

## 3.5 数据规范化（不讲）

## 3.6 状态转换图

## 3.7 其他图形工具

## 3.8 验证软件需求

## 3.2 获取需求的方法

---

- 深入面谈
- 问卷调查
- 会议
- 典型用户和典型场景
- 观察商业过程和工作流
- 原型法
- 文档分析
- 建模
- .....

## 3.2 获取需求的方法

---

### 四、典型用户和典型场景

□ 开发一个软件时，我们都知道要为用户考虑，但是**用户在哪里？不同用户的需求有没有区别？有多少不同类型的用户？**

- 在设计软件的过程中，设计/开发者往往会以自己使用产品的习惯和对软件行业的熟悉程度出发设计，忘记了软件是给千千万万个不那么会用电脑的人使用的。在这种情况下，“**典型用户**”会强迫我们在考虑问题时**从用户的角度出发**。

# Visual Studio 2005 设计阶 段使用的几个 典型用户

典型用户 的名号和造型	特点
Mort: 	<p>Mort: 不一定是专业出身的程序员，他们有自己的主业（例如统计工作者、生物信息工作者、股票分析师），编程只是一个工具，他们的主要目的是用工具把事情搞定就行了。他们很喜欢代码示例，也不特别关心程序效能。（例如，许多 VB 用户，偶尔用 VS 写程序处理数据的研究人员等。）</p> <p>最适合的任务：通过 VB/JavaScript 等去探索，把事情做出来就好了。</p> <p>做同样工作的不同视角：例如，如果要写一个用户界面的控件，这类开发人员会通过编程工具去找到一个合适的控件，简单实验后，就宣告大功告成了。</p> <p>对 IDE 的最大要求：支持所见即所得的拖动，隐藏细节。</p>
Elvis: 	<p>Elvis: 以编程为生的程序员，他们大多是计算机或软件专业出身。他们至少熟知一门编程语言，比较关注算法和程序的效能、内存泄漏等。各种 IT 公司的开发人员应该是属于这一类型。</p> <p>最适合的任务：写一组函数实现高效的数据存储和搜索。</p> <p>做同样工作的不同视角：例如，如果要写一个用户界面的控件，这类程序员会更主动地去探究控件的各种属性，控件之间的关系，写一些额外的代码，把各个控件的行为结合起来。</p> <p>对 IDE 的最大要求：支持充分自由的定制，各个技术细节要能找到详细的技术文档。</p>
Einstein: 	<p>Einstein: 在行业里战斗了很多年的程序员、架构师和非常了解技术的项目经理。他们能决定项目用什么样的技术以及发展路线。</p> <p>最适合的任务：写一个脚本语言的编译器。</p> <p>做同样工作的不同视角：例如，如果要写一个用户界面的控件，这类程序员会深入了解控件实现的细节，如何扩展这个控件去适应不同的需求，或者干脆自己重构一个控件。</p> <p>对 IDE 的最大要求：把所有细节都展现出来，所有命令都要能支持命令行的操作。</p>

## 3.2 获取需求的方法

---

□ 如何定义典型用户？首先要定义用户角色。

□ 示例：

- **受欢迎的典型用户**——指那些按设计者的期望使用系统的用户，如“网站的购物者”。
- **不受欢迎的典型用户**——指有不正当目的的用户，如在房地产业主论坛中滥发房屋中介广告的用户——但这些用户也许在别的系统（如房屋中介论坛）中是受欢迎的。

此外，还要和这些典型用户的代表交流，理解用户，理解他们的工作方式和需要。然后再修改，细化典型用户。

## 3.2 获取需求的方法

### 从典型用户到场景

- 对每个目标，列出达到目标所必须经历的过程，就是**场景**，也可以叫**故事**。有些场景描述了成功的结果，有些场景描述了失败的结果。用户和系统有各种可能的交互情况，写场景时要有针对性。
- 示例：现实生活中一个银行从业者发的一条微博，他体会了“ATM无卡取现”功能的强大：

特意带上手机和令牌不带银行卡，感受一下我行ATM的无卡取现，结果连自助银行的门儿都没进去！

**不刷卡怎么开门？？？**

## 3.2 获取需求的方法

---

### □ 购物网站卖家A上货场景示例：

- 1、A登录网站，如果A设置了“记住我的登录信息”，网站会自动登录；
- 2、A点击“上传商品信息”，进入上传页面。页面布局和最终用户看到的一样，这样在编辑时A就知道效果了；
- 3、A可先上传图像文件，网页可自动开启后台程序处理图像文件的上传；
- 4、A依次输入商品的名字、描述等。网页自动记住了他以前输入的资料，在各个字段中都有提示，他一般选中以前的输入，然后稍作修改即可；

## 3.2 获取需求的方法

---

5、A输入必填资料后，就可以选择下面两个动作之一：

- a. 立即发布
- b. 保存，不发布
- c. 保存，继续编辑

6、卖家选择a，网页会检查输入的完整性，必要时给予提示，所有资料上传到网站后，网站会自动生成上传图像的各种缩略图（64×64、128×128、512×512等），并将该产品标注为“新产品”。

**从场景到任务**



## 3.2 获取需求的方法

---

□ 按子系统/模块的所属关系划分场景为不同的任务

■ 上例的用户上货场景，可分为以下几项。

### 1. UI层

子任务：界面设计，货物资料处理，文件上传处理等。

### 2. 逻辑层

子任务：用户输入字段合法性处理，上传图像逻辑和缩略图处理，资料保存逻辑等。

### 3. 数据库

子任务：资料读取的存储过程，图像的索引建立和维护等

# 注意事项

---

## □ 用户需求获取是一门倾听的艺术

- 倾听和引导，使得客户的回答给出有价值的信息
- 协助和鼓励客户表述他们需求的艺术

## □ 比谁更聪明？

- 不要尝试向客户证明你是聪明的
- 抓住所有的机会表现你认为客户是聪明的

## □ 采用合适的需求获取技术

- 只采用一种获取技术是不够的
- 获取技术的选择与项目参与人相关
- 与待理解的需求相关，与具体应用领域相关

## 3.3 分析建模与规格说明

---

**需求分析是对应用问题及环境的理解和分析，  
为问题涉及的信息、功能及系统行为建立模型  
，将用户需求精确化、完全化，最终形成下一  
步的需求规格说明书。**

需求分析的**核心**在于**建立分析模型**

准确定义未来系统的目标，确定为了  
满足用户的需求系统必须做什么

## 3.3 分析建模与规格说明

---

### 一、分析建模

**模型是为了理解事物而对事物作出的一种抽象，是对事物的一种无歧义的书面描述。**

**为了开发复杂的系统，应从不同角度(模型)抽象出目标系统的特性(数据模型、功能模型、行为模型)**

# 分析模型的描述工具

采用结构化、模块化、自顶向下、由外到内、逐层分解的思想分析系统。

将数据和逻辑结构抽象为对象，运用对象属性和方法来处理业务数据和逻辑。

	面向过程的需求分析	面向对象的需求分析
数据模型	实体-联系图 ( ERD ) 数据字典 ( DD )	类图、类关系图
功能模型	数据流图 ( DFD )	用例图
行为模型	状态变迁图 ( STD )	活动图、时序图、状态图

# 1、面向过程分析模型——结构化分析方法

---

- 面向**数据流**进行需求分析的方法
- 结构化分析方法适合于**数据处理类型软件**的需求分析
- 具体来说，**结构化分析方法**就是用抽象模型的概念，按照软件内部**数据传递、变换**的关系，**自顶向下**逐层分解。

# 功能模型——数据流图 (书2.4)

- 从数据传递和加工的角度，以图形的方式刻画数据流从输入到输出的移动变换过程。
- 数据流图中的基本符号



## 1、数据流

- 是数据在系统中的传送通道，数据流符号的箭头指明了数据的流动方向，由一组固定成分的数据组成，如“选课单”由“学号、姓名、课程编号、课程名称”等成分组成。
- 数据流可**从加工流向加工**，也可在**加工与数据存储或外部项之间**流动；两个加工之间可有多**股数据流**
- 数据流的命名
  - 用名词，不要使用意义空洞的名词
  - 尽量使用现实系统已有名字

请自学P41图2.4  
数据流图的符号



# 功能模型——数据流图

## 画数据流时需注意的问题：

### 1、不要把控制流作为数据流

如：下图中读下张卡属于控制流，不应画出。



### 2、不要标出激发条件



## 2、加工

- 表示对数据进行的操作，如“处理选课单”等
- 每个加工至少有一个输入数据流和一个输出数据流
- 加工的编号，说明这个加工在层次分解中的位置

(分层DFD)

- 加工的命名

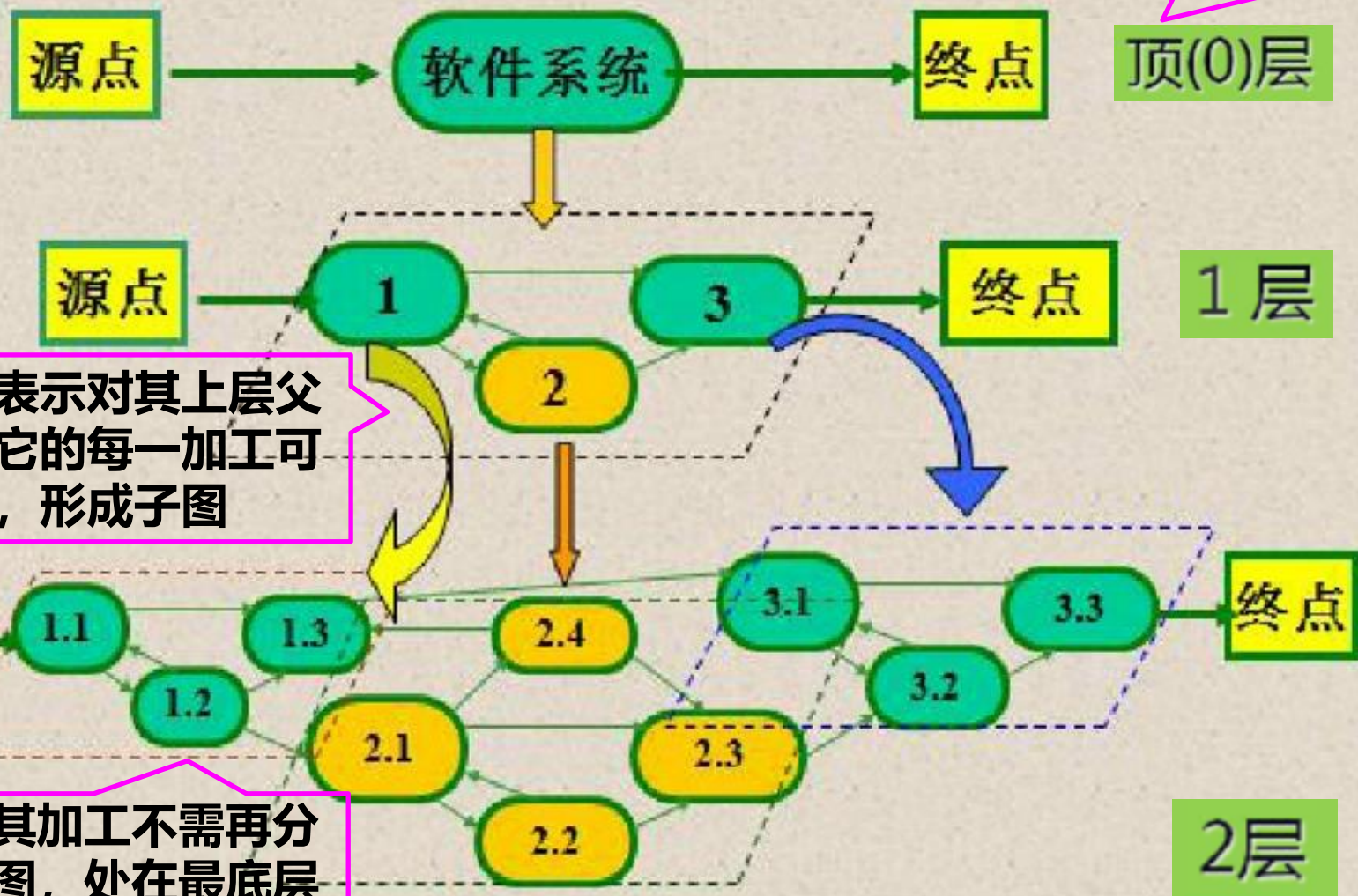
- 顶层的加工名就是整个系统项目的名字
- 尽量最好使用动宾词组，也可用主谓词组
- 不要使用空洞的动词



# 功能模型——数据流图

## 分层的数据流图

顶层流图仅包含一个加工，代表被开发系统



中间层流图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

底层流图指其加工不需再分解的数据流图，处在最底层

## 3、数据存储

- 用于保存数据的数据文件，它可以是数据库文件或任何其他形式的数据组织。
- **流向数据存储的数据流**可理解为向文件写入数据或对文件进行查询。
- **流出数据存储的数据流**可理解为从文件中读取数据或得到查询结果。

## 4、外部实体（数据源点/终点）

- 反映数据流图与外部实体之间的联系，表示图中的**输入数据来自哪里或处理结果送向何处**。

# 数据流图的建立方法

---

按照结构化分析方法中“**自顶向下，逐步分解**”的思想。

## (1) 建立顶层数据流图

- 先将整个系统看作是一个加工，它的输入数据和输出数据表明了系统和外部环境的接口。
- 任何系统的顶层流程图都只有一个。

## (2) 数据流图的分层细化

- 从顶层数据流图出发，逐层地对系统进行分解。
- 每分解一次，系统中加工的数量就随之增加，每个加工的功能描述也越来越具体。
- 重复这种分解，直至得到系统的底层数据流图。

# 示例：仓库管理与订货系统

仓库业务的工作过程：**企业职工**填写**领料单**，到仓库领取零件。**仓库管理员**审查领料单是否有效正确，不正确的领料单退还职工，填写正确的领料单则办理领料手续，填写**登记单**，修改库存量并给予零件。当某种零件的库存量低于事先规定的临界值时，登记需要采购零件的订货信息，为**采购员**提供一张**订货单**。

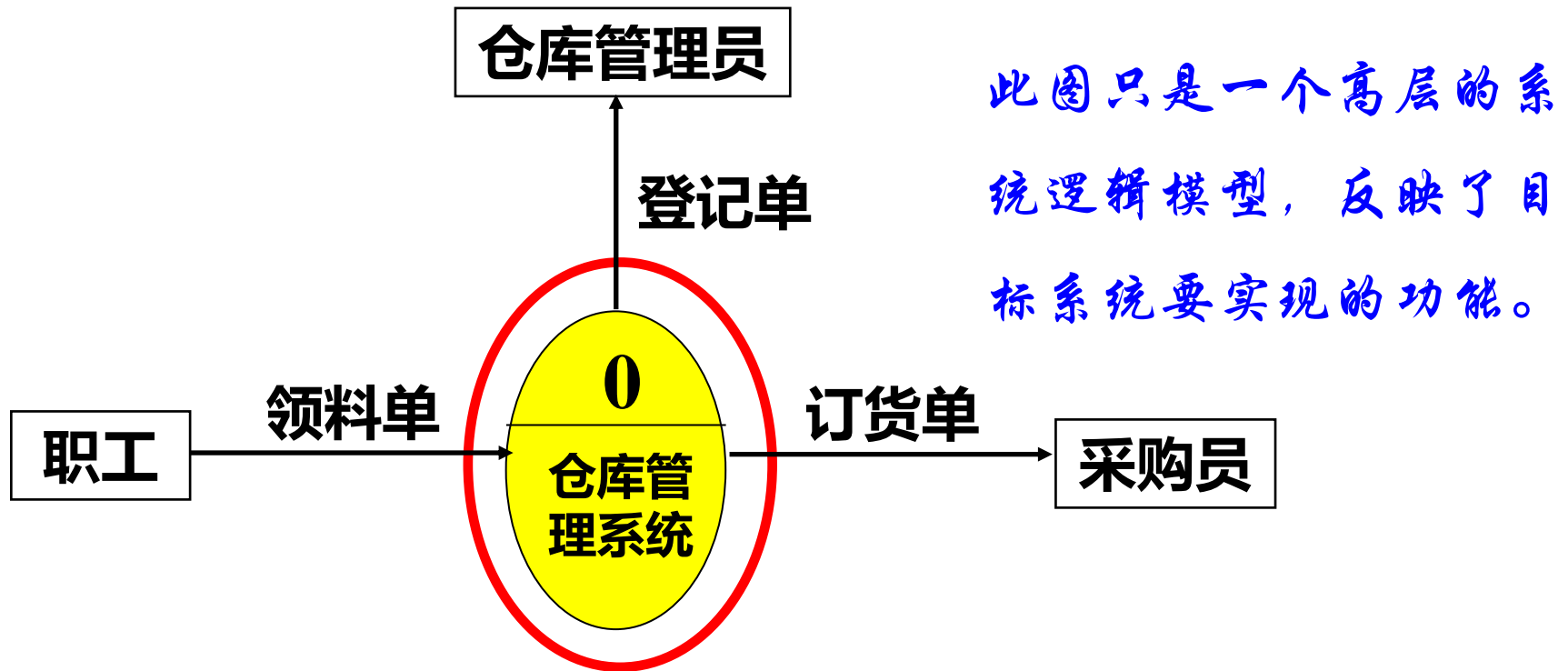
先分析源点、终点和输入输出数据流

■ 源点：职工

■ 终点：仓库管理员、采购员

■ 数据流：领料单、登记单、订货单

# 第一步，绘制顶层数据流图



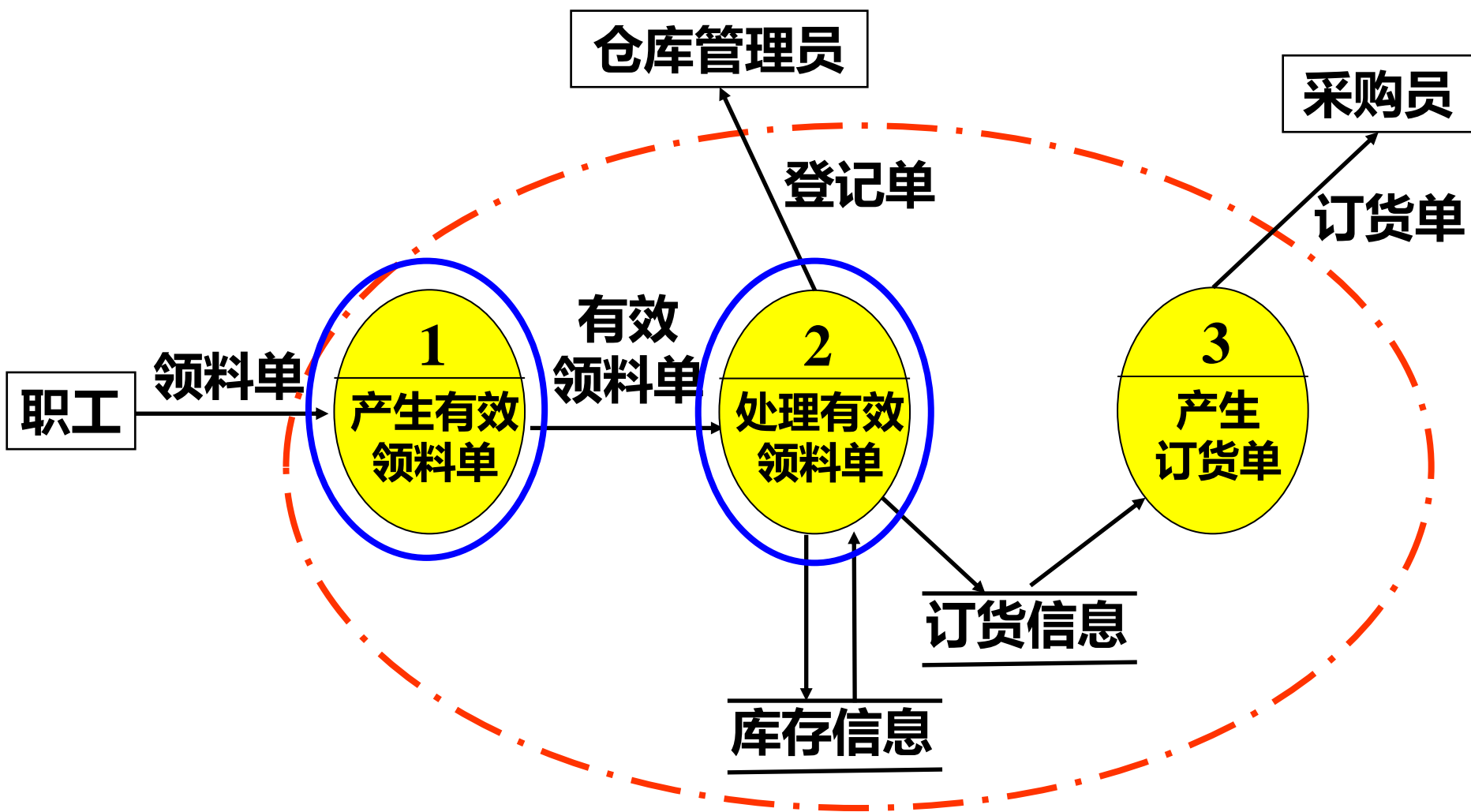
## 顶层(0层)数据流图

- 数据存储：本图中没有。



## 第二步，绘制数据流图1层

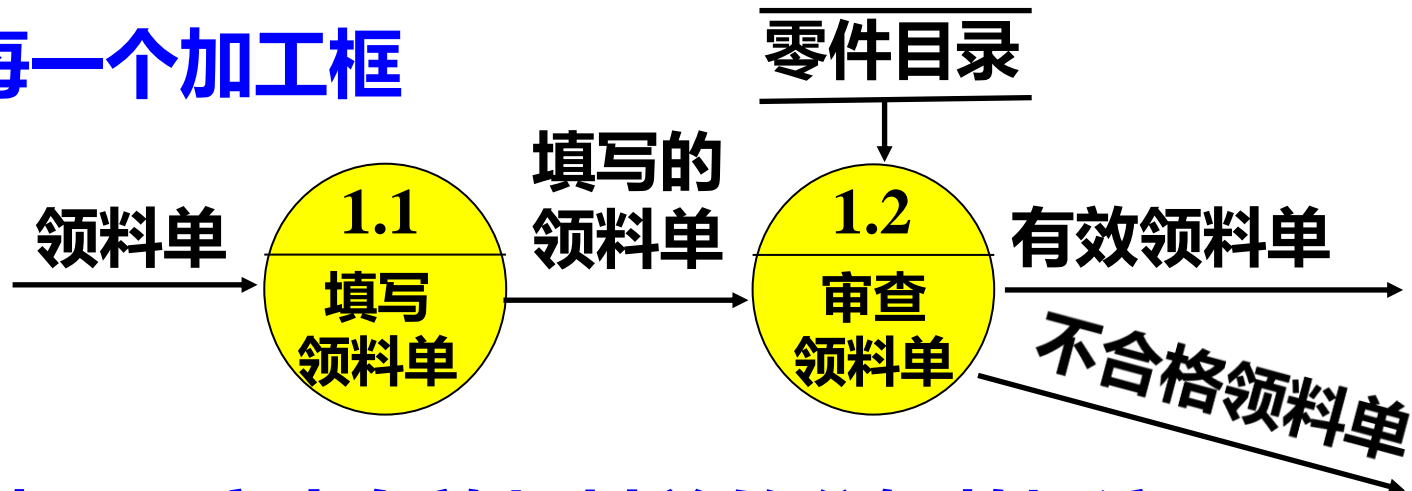
从输入端开始，根据仓库业务工作流程，画出数据流流经的各加工框，逐步画到输出端，得到1层数据流图。



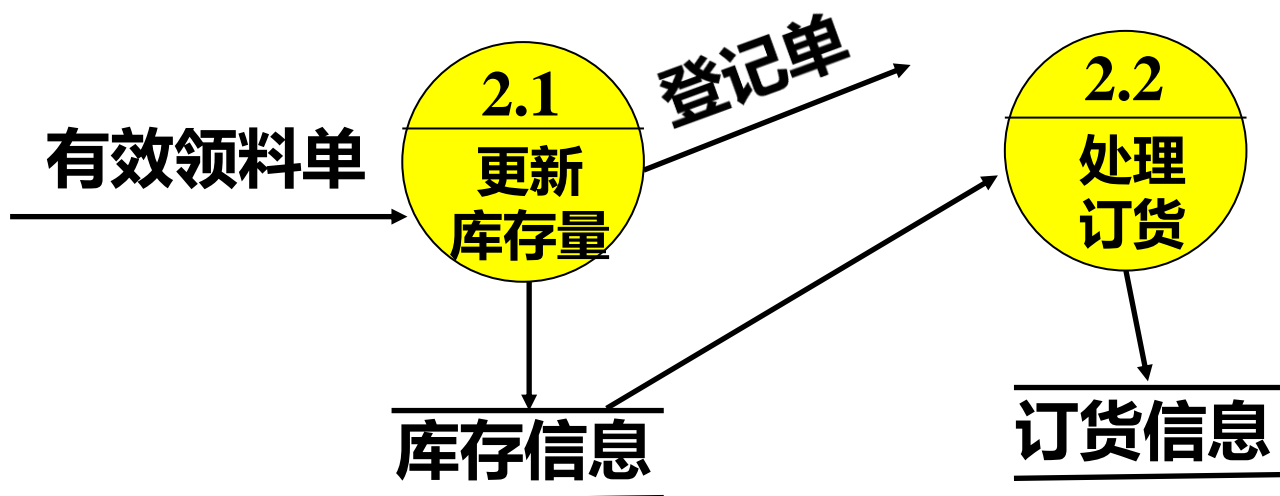


## 第三步，绘制2层数据流图

### 加细每一个加工框

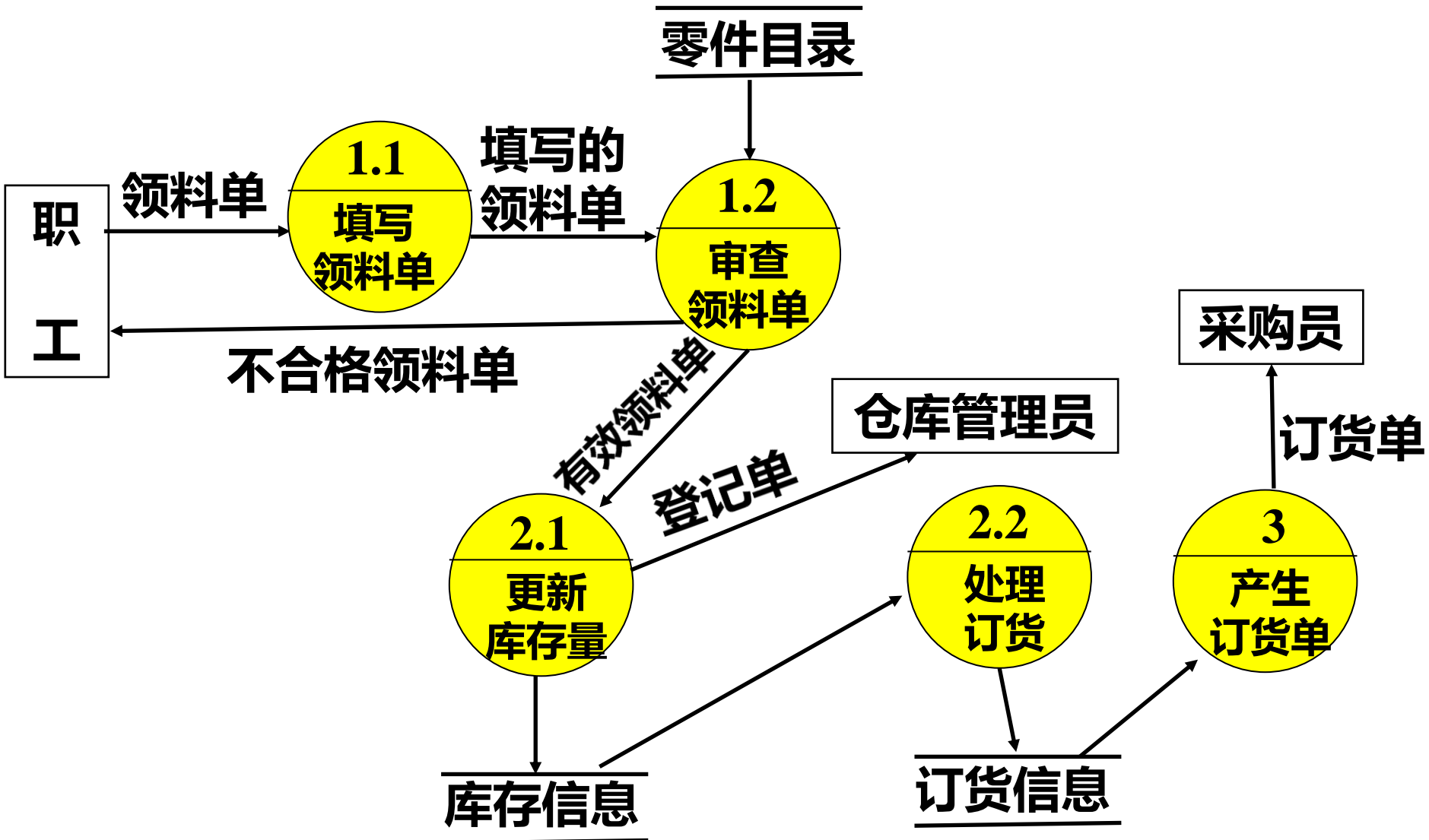


### 加工1产生有效领料单的分解数据流图



### 加工2处理有效领料单的分解数据流图

## 第四步，合成总体数据流图



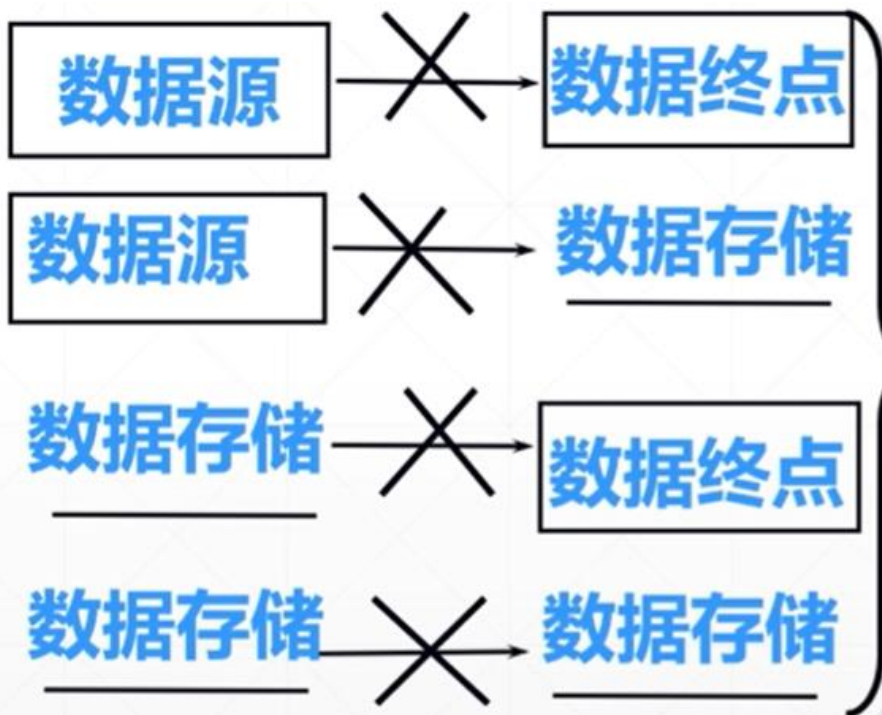
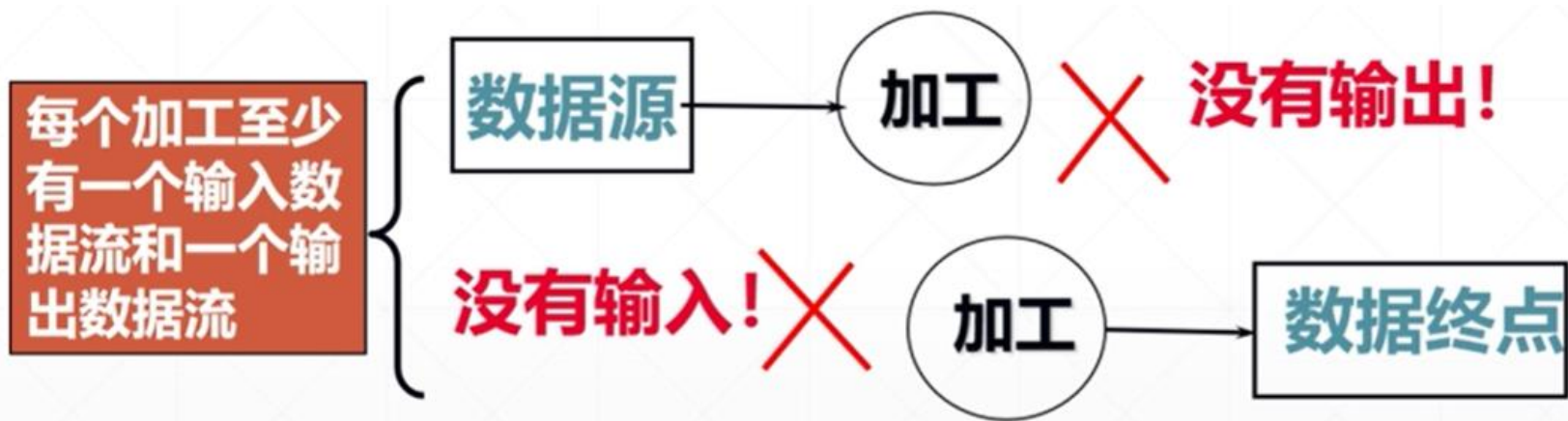
合成的仓库管理系统总数据流程图

## 第五步，检查与调整数据流图

---

- 在分析过程中，由于每个人的经验和思路不尽相同，对数据流图的分解方案可以有多种形式，不是唯一的。**对每一张数据流图进行检查**，如果太不均衡，就需要进行调整，尽量使分解后的各个软件子系统的复杂性得到均衡，便于今后设计工作的并行开展。

# 数据流图中的常见错误



数据流必须要么从某个加工流出、要么流入某个加工，而不能直接从外部项流向数据存储等等。图示的几种流动都是不合理的

# 检查数据流图的原则

---

- 1) 所有图形符号只限于前述四种基本图形元素
- 2) 主图必须包括前述四种基本元素，缺一不可
- 3) 每个加工至少有一个输入数据流和一个输出数据流
- 4) 需按层给加工框编号。编号表明该加工所处层次及上下层的亲子关系
- 5) 任何一个数据流子图必须与它上一层的一个加工对应，两者的输入数据流和输出数据流必须一致。此即父图与子图的平衡
- 6) 图上每个元素都必须有名字
- 7) 数据流图中不可夹带控制流
- 8) 初画时可以忽略琐碎的细节，以集中精力于主要数据流

- **数据字典(Data Dictionary)是数据流图的补充,能够准确地定义数据流图中各组成成分的具体含义**
  - **数据字典用简洁、清晰、易理解的文字描述条目,说明数据流图的加工功能、性能、要求及设计约束等。**
  - **数据流图和数据字典共同构成了系统的功能逻辑模型。**
  - **一般说来, 数据字典应该由对下列4类元素的定义组成:**
    - (1) **数据流**
    - (2) **数据流分量(即数据项)**
    - (3) **数据存储**
    - (4) **处理**

# 数据流条目

---

- 一般应包括以下内容：数据流名称、数据流别名、说明、数据流来源、数据流流向、数据流组成和数据流量等。

例如：仓库管理系统中的领料单数据流在数据字典中的条目描述

- 数据流名称：领料单
- 数据流别名：无
- 说明：职工填写的领料单
- 数据流来源：职工
- 数据流流向：加工1.2(审查领料单)
- 数据流组成：材料名称+材料规格+领用数量+日期+  
领用人工号+领用人姓名+领用单位
- 数据流量：5份/小时

□ 数据流图中每个数据结构都由若干个数据项构成的。

■ 数据项名称、数据项别名、说明、类型、长度、取值范围及含义等。

例如：领料单中的工号数据项在数据字典中的条目描述为

- 数据项名称：工号
- 数据项别名：employee\_no
- 说明：本单位职工的惟一标识
- 类型：字符串
- 长度：6
- 取值范围及含义：1 ~ 2位(00..99)为部门编号：  
3 ~ 6位(XX0001..XX9999)为人员编号



# 数据文件条目

---

□ 数据文件条目中应包含的主要内容有：

■ 数据文件名称、说明、数据文件组成、组织方式、存取方式、存取频率等。

例如：仓库管理系统中的库存信息文件在数据字典中的条目描述为

- 数据文件名称：库存信息
- 说明：存放材料的库存信息
- 数据文件组成：材料编号+材料名称+材料规格+材料类别  
+厂家+货架号+库存量+备注
- 组织方式：按材料编号从小到大排列
- 存取方式：顺序
- 存取频率：5次/时

# 数据加工条目

---

数据字典的数据文件条目中应包含的主要内容有：

- 数据加工名称、加工编号、说明、输入数据流、输出数据流、加工逻辑等。

例如：仓库管理系统中的“审查领料单”加工在数据字典中的条目描述为

- 数据加工名称：审查领料单
- 加工编号：1.2
- 说明：根据职工的填写的领料单审查其是否合法
- 输入数据流：领料单
- 输出数据流：有效领料单
- 加工逻辑：根据职工信息及库存信息审核该领料单是否合法

**假设某信息系统有如下功能：**

**(1) 计算标准分：根据考生原始分计算，得到标准分，存入考生分数文件。**

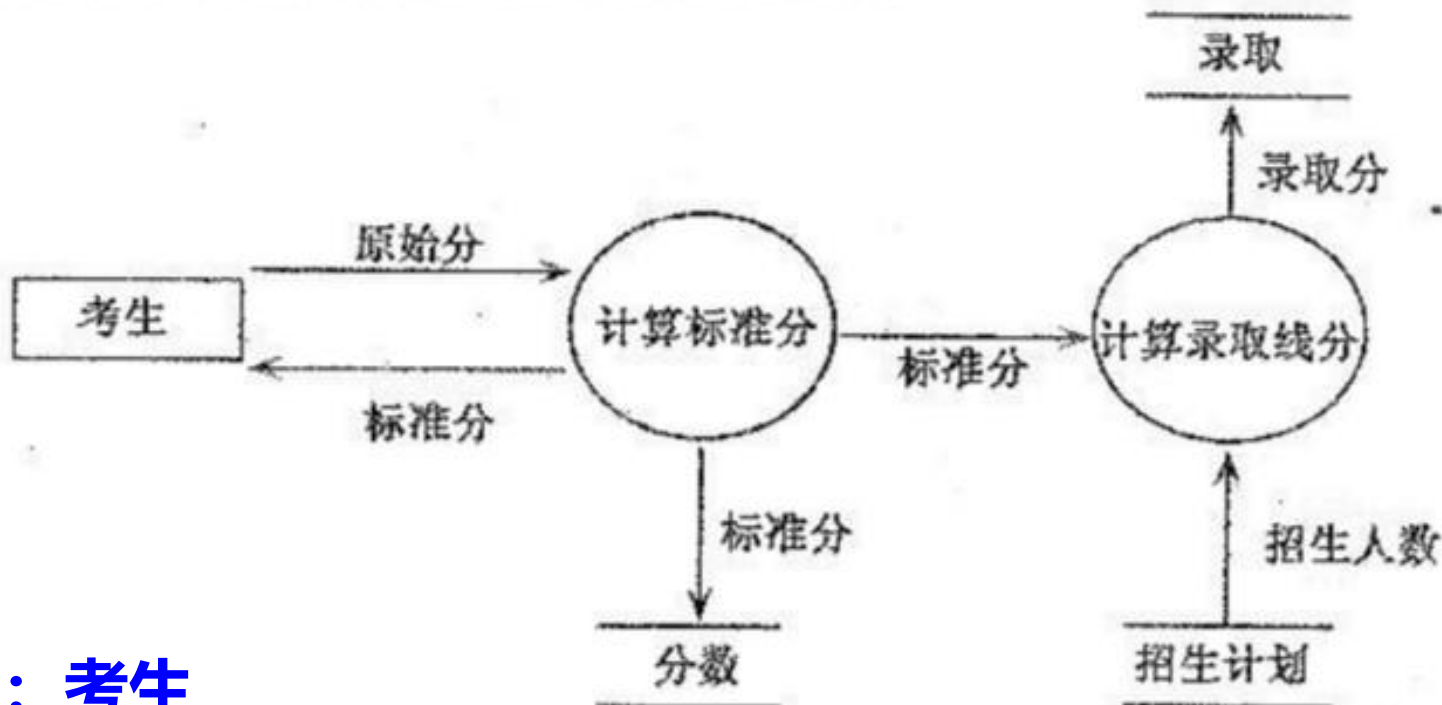
**(2) 计算录取线分：根据标准分、招生计划文件中的招生人数，计算录取分，存入录取文件**

**请画出该系统的数据流图，并指出图中各组成部分包含的元素。**

作答

# 课堂作业

- 1、计算标准分：根据考生原始分计算，得到标准分，存入考生分数文件。
- 2、计算录取线分：根据标准分、招生计划文件中的招生人数，计算录取分，存入录取文件。



**源点和终点：考生**

**加工：计算标准分、计算录取线分**

**数据流：原始分、标准分、录取分、招生人数**

**存储文件：录取文件、分数文件、招生计划文件**