北京交通大學
BEIJING JIAOTONG UNIVERSITY

软件系统分析与设计
System Analysis & Design

M210007B [03]

Haiming Liu

Tuesday, April 19, 2022

课程计划

软件工程的基础

1. 软件工程基础/软件生命周期
2. 软件过程模型
3. 软件项目管理

**系统分析与需求描述
结构化设计方法**

4. 系统分析方法与问题定义
5. 需求分析与需求获取
6. 用例建模与用例描述
7. **结构化系统分析与设计**

面向对象的设计方法

8. 面向对象方法概述
9. 面向对象建模与UML
10. 软件系统动态建模
11. 软件系统静态建模

12. 数据库设计
13. 综合案例分析
14. 考前复习

系统分析方法
System Analysis Approaches

- 模型驱动分析法 Model-Driven Analysis
- 加速系统分析法 Accelerated Systems Analysis
- 需求获取法 Requirements Discovery
- 业务过程重构法 Business Process Redesign
- *FAST* 系统分析策略 *FAST* Systems Analysis Strategies

模型驱动分析法
Model-Driven Analysis Approaches

- 模型驱动分析：一种解决问题的方法，强调绘制图形系统模型以记录和验证现有和提议的系统。 最终，系统模型成为设计和构建改进系统的蓝图。

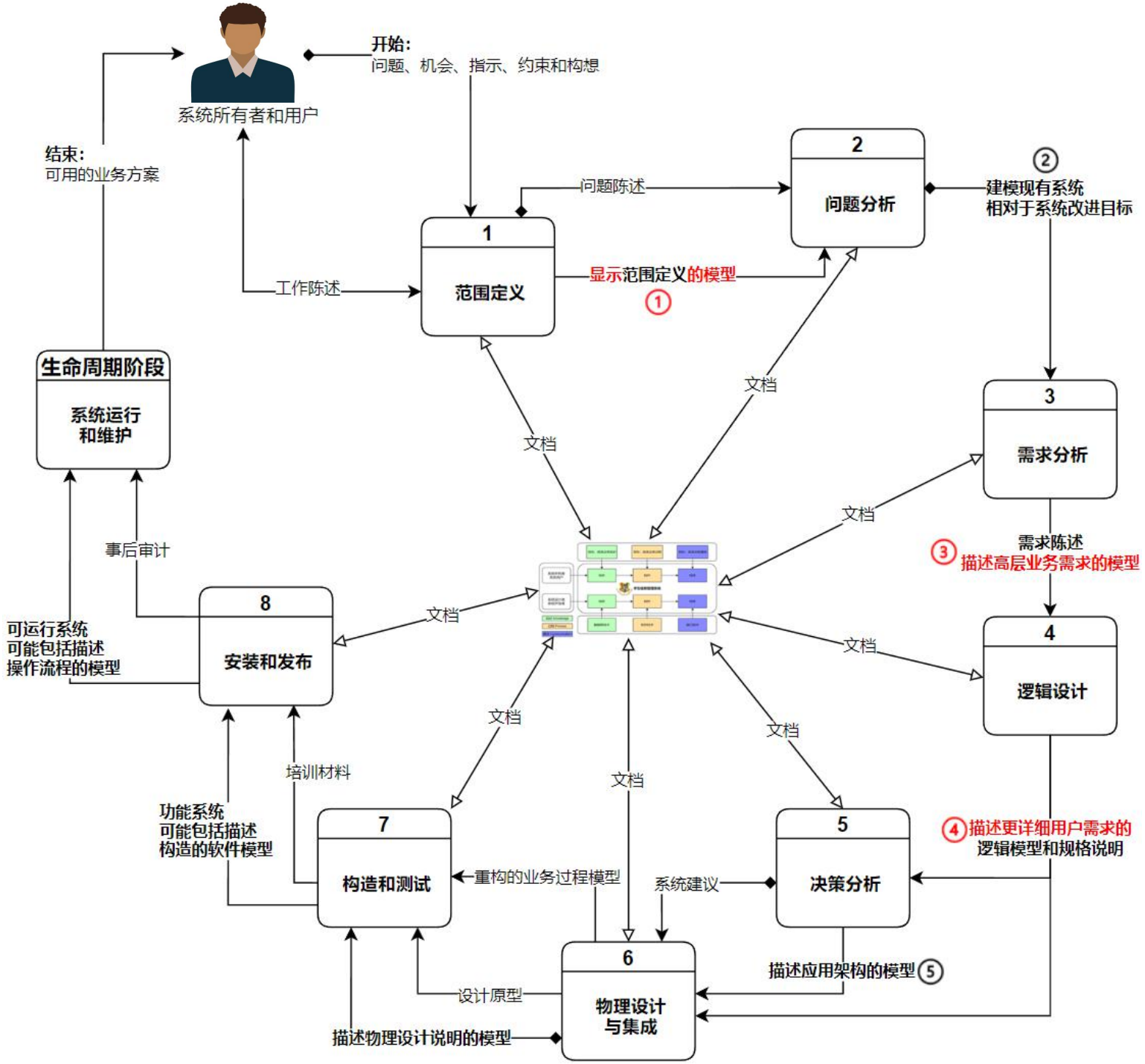
- **Model-driven analysis** – a problem-solving approach that emphasizes the drawing of pictorial system models to document and validate both existing and/or proposed systems. Ultimately, the system model becomes the blueprint for designing and constructing an improved system.

什么是模型?
What is a Model?

- 模型：对于某个实际问题或客观事物、规律进行抽象后的一种形式化表达方式。“一图抵千言”，大多数模型都用图形来进行交流与分析。

- **Model** – a representation of either reality or vision. Since “a picture is worth a thousand words,” most models use pictures to represent the reality or vision.

系统开发过程视图


开始:
问题、机会、指示、约束和构想
系统所有者和用户
结束:
可用的业务方案
2
问题分析
问题陈述
建模现有系统
相对于系统改进目标
1
范围定义
显示范围定义的模型
①
工作陈述
生命周期阶段
系统运行
和维护
文档
文档
3
需求分析
文档
需求陈述
③ 描述高层业务需求的模型
事后审计
8
安装和发布
可运行系统
可能包括描述
操作流程的模型
文档
文档
4
逻辑设计
文档
文档
培训材料
文档
功能系统
可能包括描述
构造的软件模型
7
构造和测试
重构的业务过程模型
系统建议
5
决策分析
④ 描述更详细用户需求的
逻辑模型和规格说明
6
物理设计
与集成
描述应用架构的模型 ⑤
设计原型
描述物理设计说明的模型

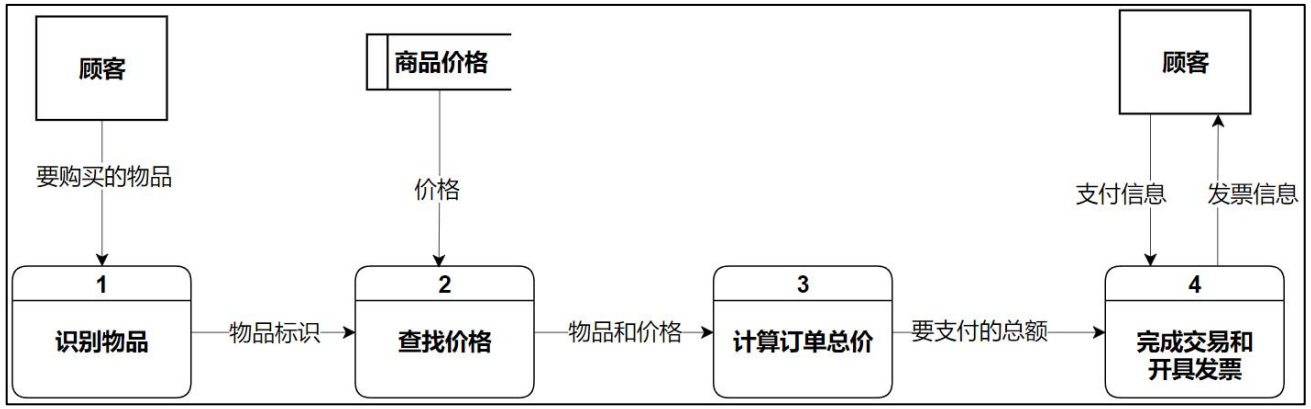
/-1 过程建模
Process Modeling

结构化系统分析与设计
Structured Systems Analysis and Design

结构化方法：对系统构造模型，作为理解系统和记录业务需求或技术设计的方法。

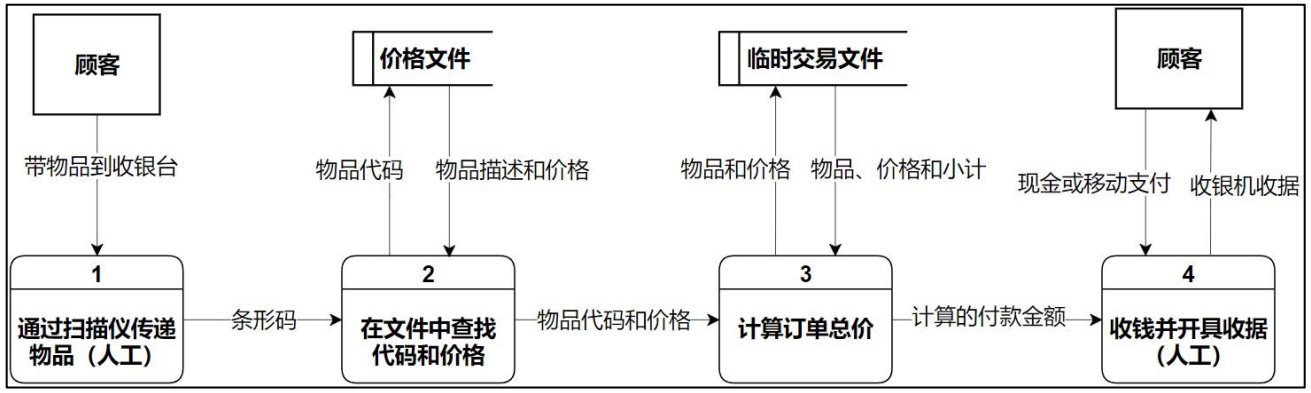
逻辑模型：

- 展示了系统是什么或者系统做什么
- 逻辑模型说明了系统的本质
- 又称本质模型、概念模型和业务模型



物理模型：

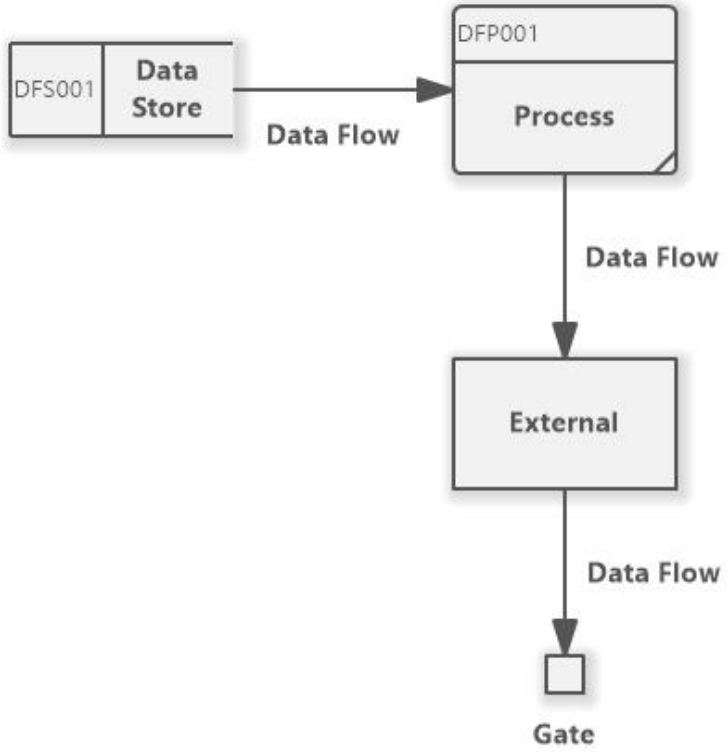
- 不仅展示了系统是什么或者系统做什么，还展示了系统技术上如何实现
- 反映了技术选择和选技术的限制



逻辑过程建模技术
Logical Process Modeling

结构化方法：

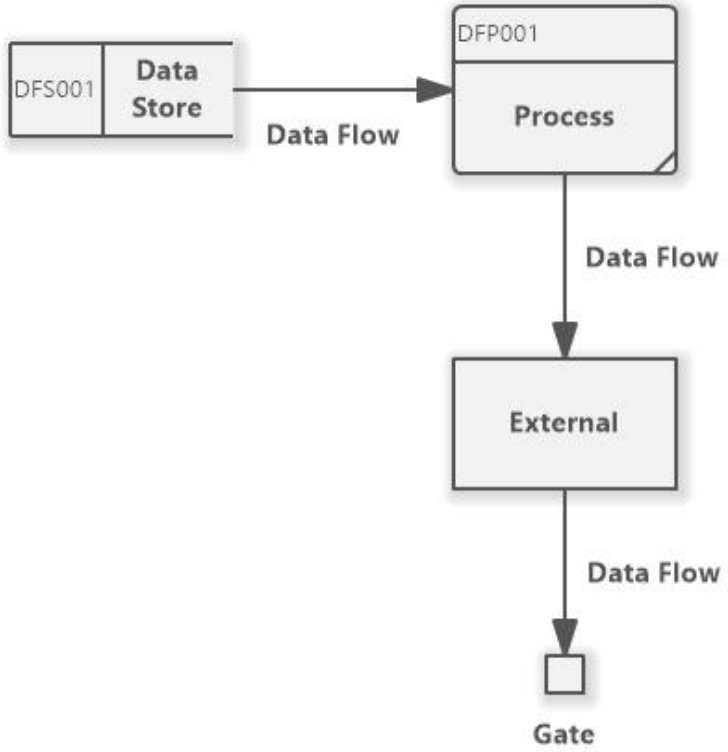
- 一种*面向数据流的需求分析和设计方法*，适用于分析和设计大型数据处理系统，简单、实用、应用广泛
- 从系统建模的视角来看，基于为*过程建模*而使用的*数据流图* (Data Flow Diagram, DFD)技术
- 系统以*过程为中心*，在功能分解活动中被分解为可管理的单元，同时将系统划分为*由数据流连接的业务过程*



过程建模
Process Modeling

过程建模：

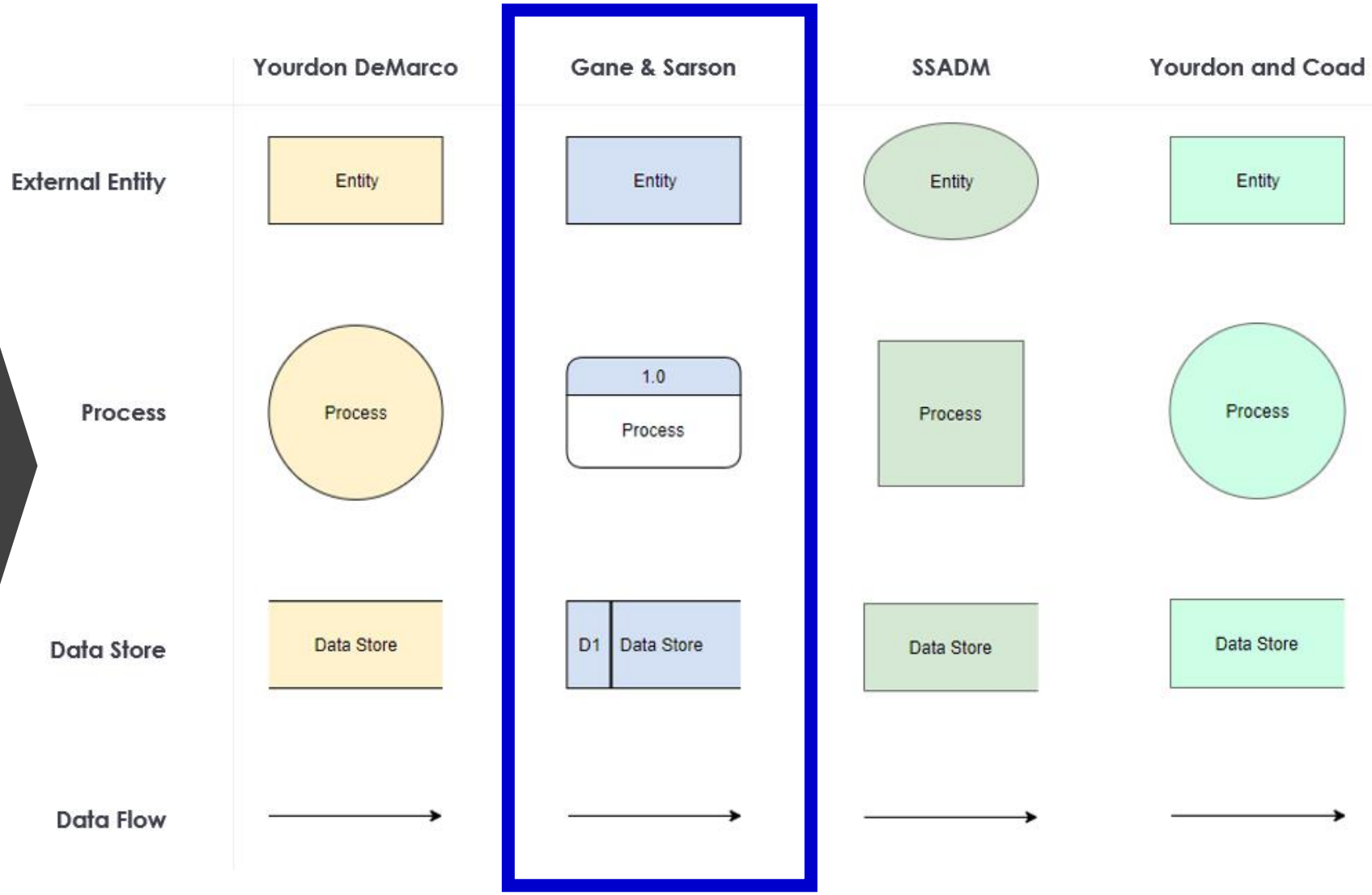
- 一种组织和记录数据的结构和流向的技术
- 记录系统的“过程”和/或有系统的“过程”实现的逻辑、策略和程序。



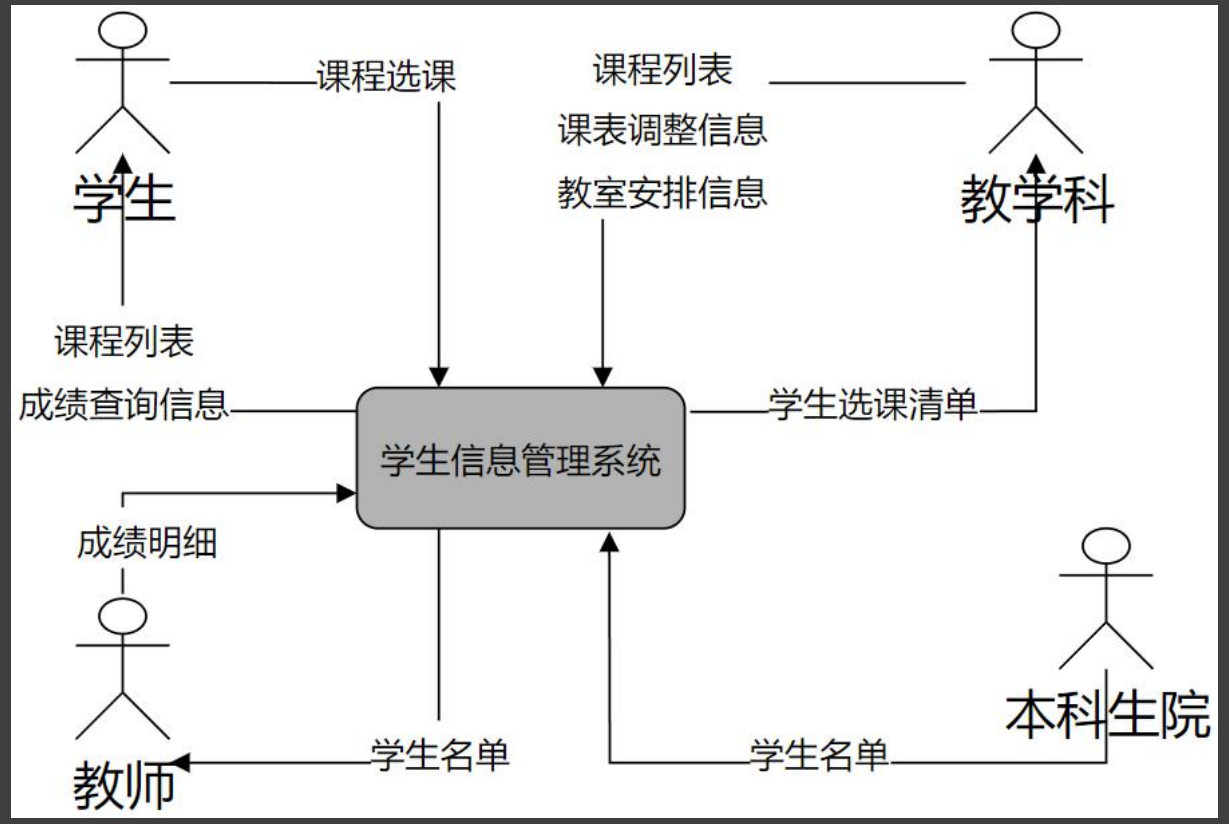
数据流图(Data Flow Diagram，DFD；又称泡式图、转换图和过程模型)：

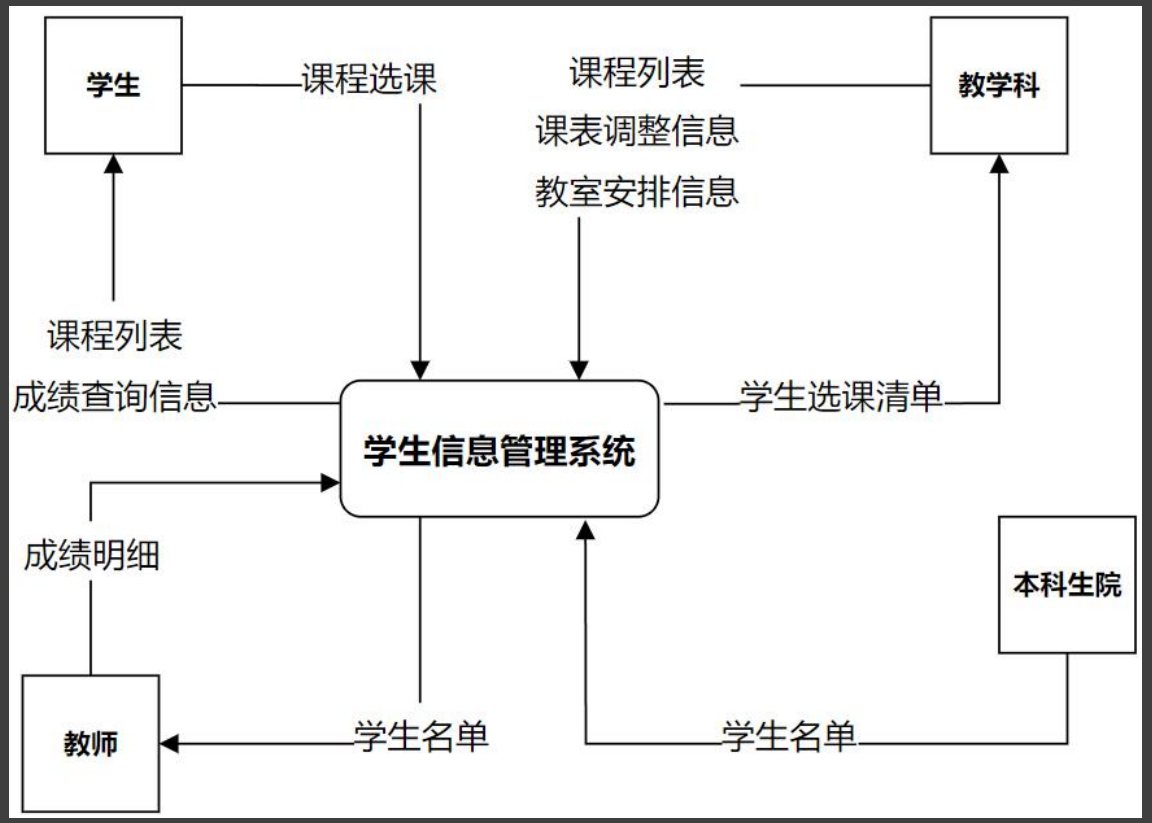
- 一种描述通过系统的数据流以及系统实施的工作或处理过程的工具
- 优点：容易阅读，仅有三种符号和一种连接

数据流图
Data Flow Diagram



DFD Notations





数据流图
Data Flow Diagram

- 圆角矩形表示要完成的过程或者工作
- 正方形表示外部代理——系统的边界
- 箭头表示数据流——或者输入和输出，到过程和来自过程
- 开放的方框表示数据存储——有时称为文件或者数据库

外部代理
External Agent

外部代理又称外部实体，定义位于项目范围之外，但与正在被研究的系统交互的人、部门、其他系统或者其他组织

- 外部代理很少是固定的。如果系统范围扩大，外部代理可能会被认为是系统内部的一个新过程
- 用正方形表示，用描述性的单数名词命名，如：学生、供应商或信息系统
- 放置在页面的周围，与它的系统边界的定义相一致
- 为避免数据流线的交叉，允许使用重复的外部代理

Gane and Sarson shape

数据存储
Data Store

数据存储，又称**文件、数据库**。是一个数据的“仓库”，收集数据供日后使用

- 描述关于企业想存储数据的“事物”，包括：
  - 人 (或一组人)：学生、部门、客户
  - 地点：销售地区、建筑物、校园
  - 对象 (或一类对象的说明)：图书、工具、软件包
  - 事件：添加、删除、修改、查看

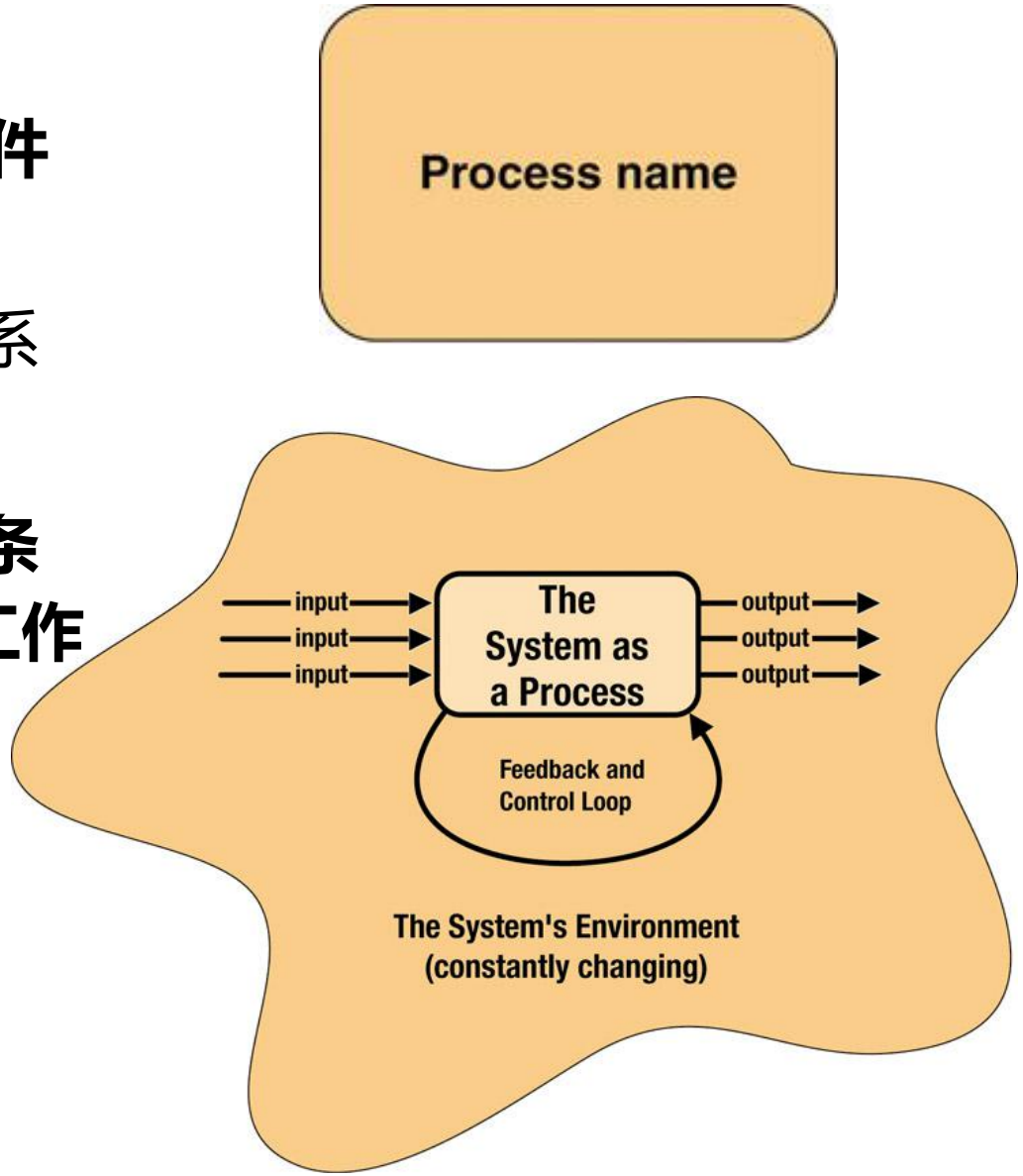
Gane and Sarson shape

- **数据存储表示了数据实体的所有具体值**
- 用末端开口的矩形（左端封闭，右端开口）表示，用名词命名
- 为避免数据流线的交叉，允许使用重复的数据存储

过程概念
Process Concept

过程是一个基本的信息系统构件。**过程响应业务事件和条件，并将“数据”转换成有用的信息。**

- 对**过程建模**帮助我们理解过程与系统环境、其他系统以及其他过程如何交互
- 使用圆角矩形表示过程，**过程是在输入数据流或条件上执行，或者对输入数据流或条件作出响应的工作**



过程概念
Process Concept

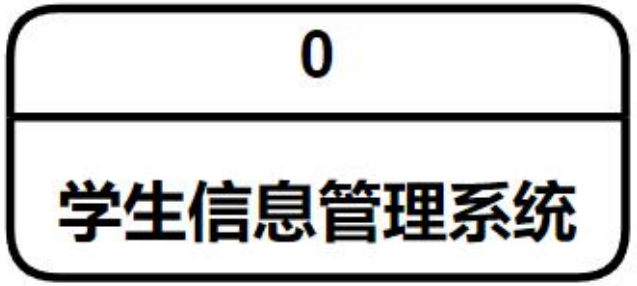
- 命名高级过程，以整个系统/子系统的名称命名

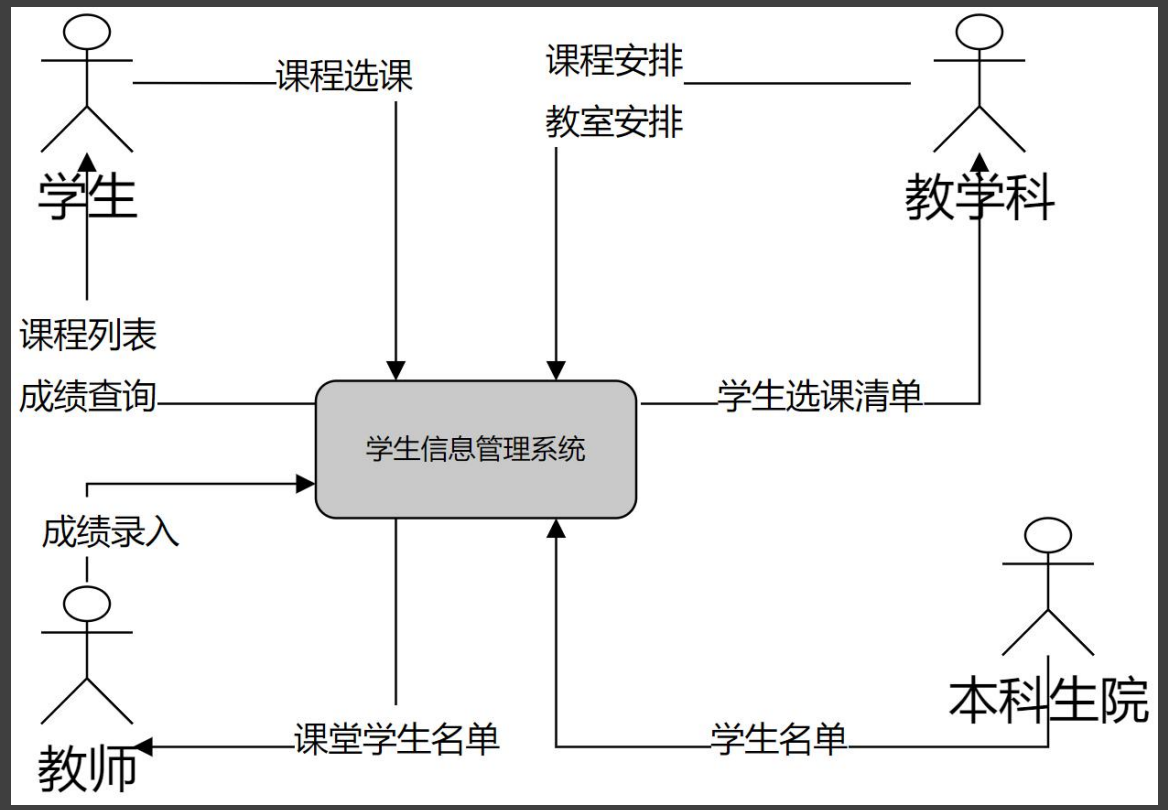
  例如：学生信息管理系统、库存管理子系统

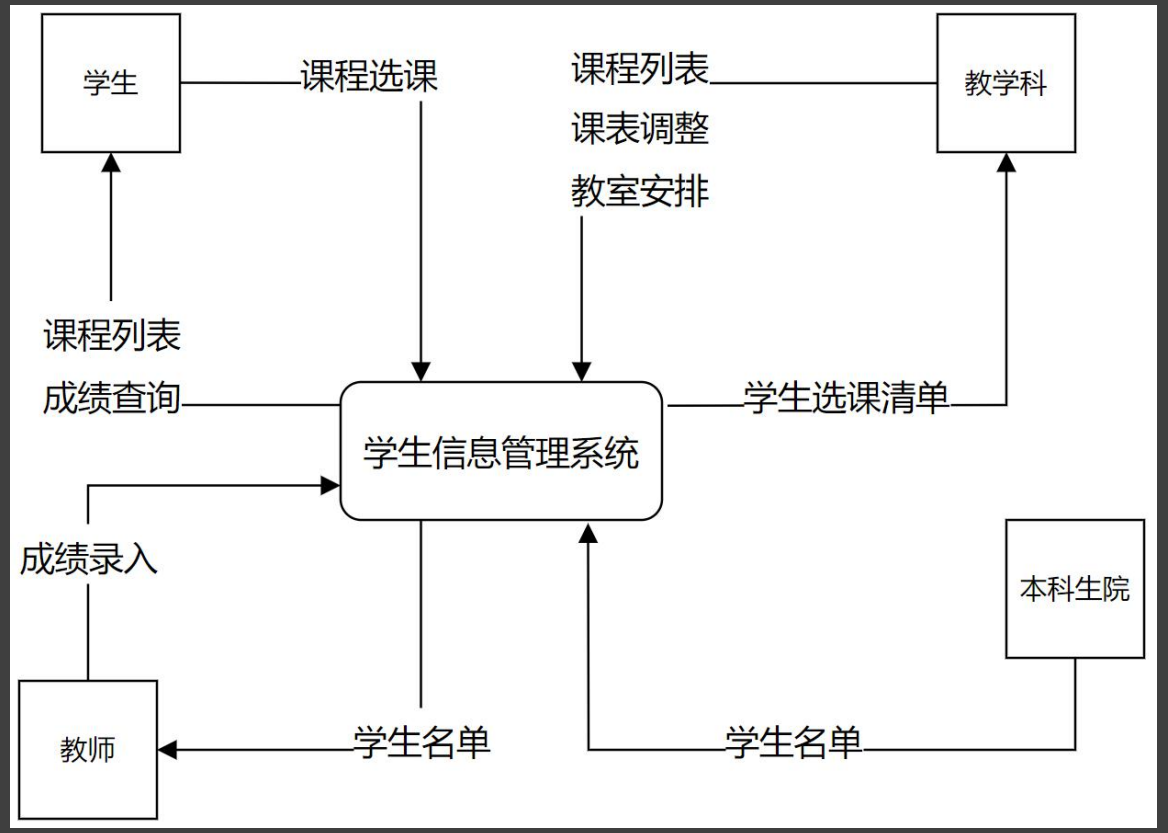
- 命名一般过程，使用“动词-形容词-名词”格式

  例如：验证账号沟的状态、添加订单的记录等

- 必须有唯一的标识号，指出它在图中的层次





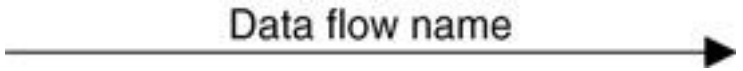


数据流图
Data Flow Diagram

- 圆角矩形表示要完成的过程或者工作
- 正方形表示外部代理——系统的边界
- **箭头表示数据流——或者输入和输出，到过程和来自过程**
- 开放的方框表示数据存储——有时称为文件或者数据库

数据流
Data Flow

数据流，表示到一个过程的数据输入，或者来自一个过程的数据（信息）输出。



- 数据流是运动中的数据
- 也用于表示在文件或数据库中创建、删除、修改或读取数据

控制流，表示触发一个过程的条件或非数据事件。

- 可以把它看作是系统工作时的一个监控条件

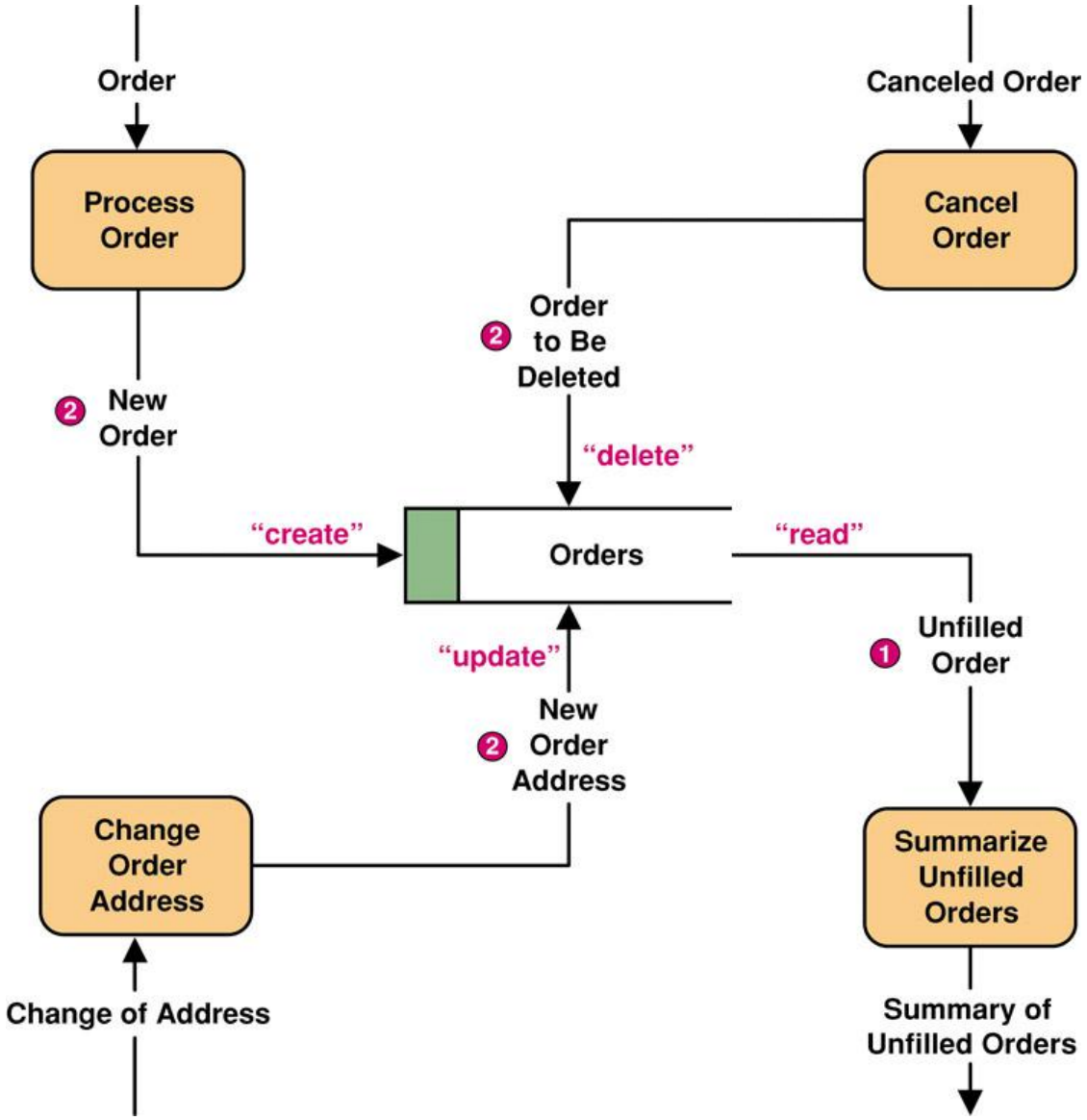
例如：时间、温度和高度等。报告生成过程可以由时序事件“每晚”、“月末”触发

- 使用描述性名词命名

数据流 Data Flows

① 从一个数据存储到一个过程的数据流，清楚地指出读取了什么数据

② 从一个过程到一个数据存储的数据流，清楚的命名以反映执行的特定动作



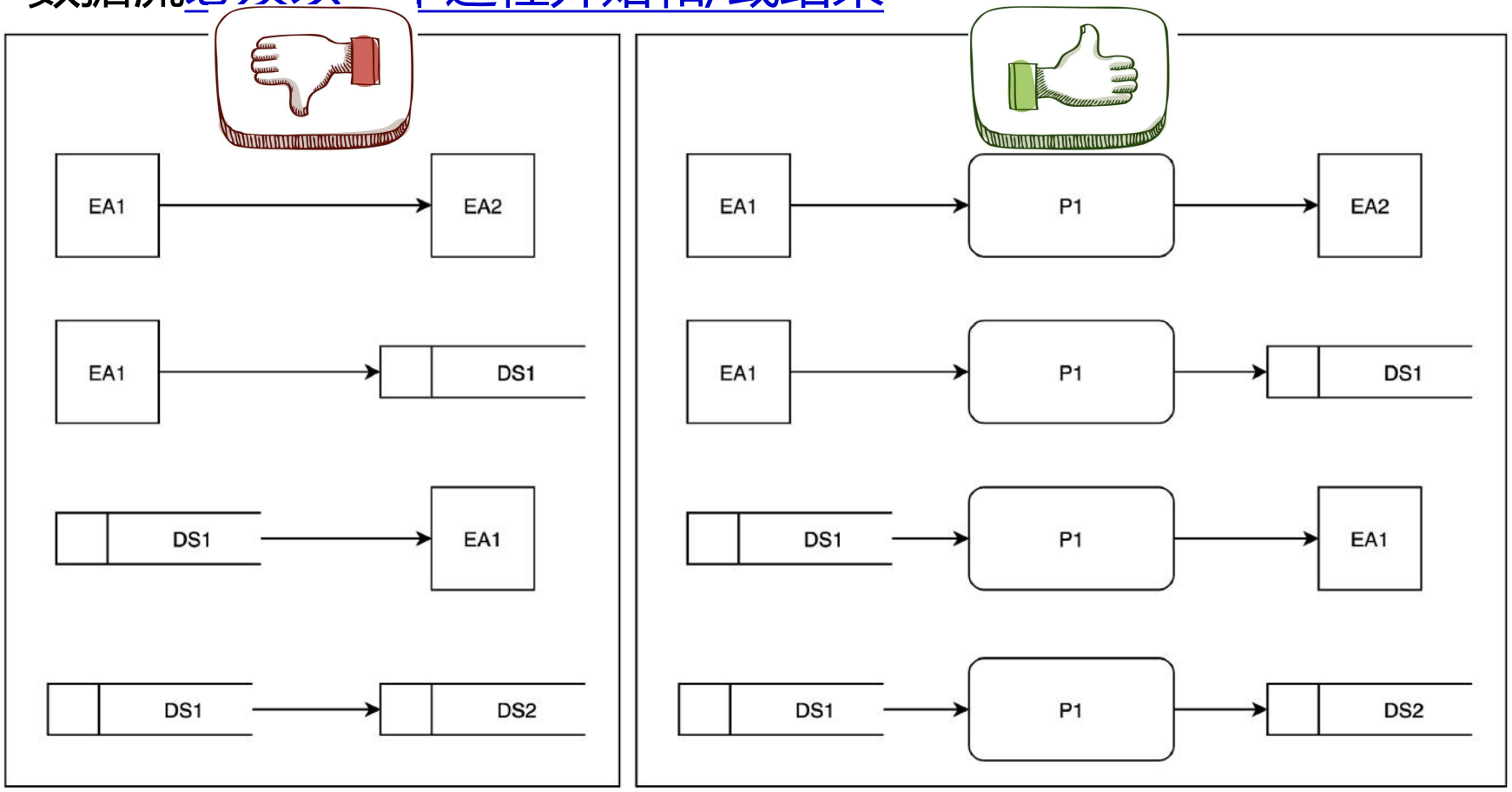
往返于数据存储的数据流

数据流规则 Rules for Data Flows

- 数据流名称应该是唯一的，并采用描述性单数名词和名词短语

例如：一个过程的输入是“订单”，输出可以命名为“有效的订单”

- 数据流名称应该描述数据而不描述实现
- 数据流必须以一个过程开始和/或结束



分支流和合并流 Divergent and Convergent Flows

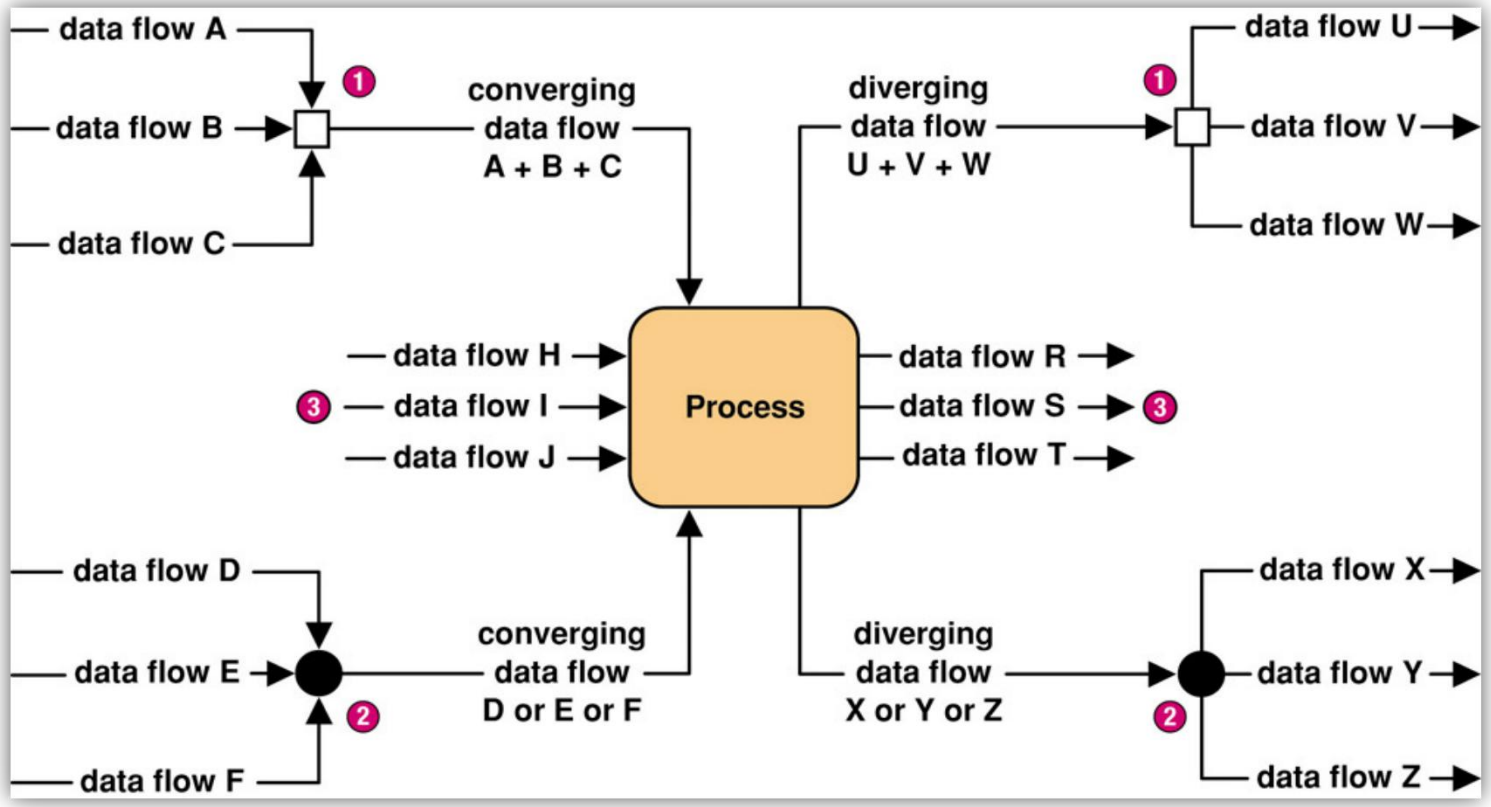
分支的数据流指示了一个数据流的所有或者部分路由到不同目的地。

合并的数据流指示了不同来源的数据流可以（必须）合并成一个报文供后续处理。

① 小方块交叉点是“与”。每次执行过程时，**必须输入（或输出）所有的分支数据流或合并数据流。**

② 小黑色交叉点是“互斥或”。每次执行过程时，**必须只输入（或输出）分支数据流或合并数据流中的一个。**

③ 在没有分支数据流或合并数据流的情况下，应假设存在一个“包含或”。每次执行过程时，**可以输入任何或者所有的数据流。**



数据流的守恒 Data Flow Conversation

数据守恒，也称“饥饿过程”，要求一个数据流只包含接收数据的过程真正需要的数据

- 重新设计业务流程以识别和消除低效率
- 简化这些过程之间的接口
- 必须精确定义每个数据流的数据组成成分，并以数据结构的形式表示

逻辑过程 Logical Processes

逻辑过程，无论如何实现这个系统都**必须实施的工作或行动**。每个逻辑过程，都由一个或多个物理过程实现。

逻辑过程的**命名**取决于过程在图中的位置，以及描述的过程类型。存在三类逻辑过程：

- **功能过程**
- **事件过程**
- **基本过程**

逻辑过程 Logical Processes

- **功能(function)：企业的一套相关的和正在进行的活动。每个功能都可能包括数个或数十个更离散的活动和过程。**
- **功能用描述整个功能的名词命名**

例如：生产系统包括以下功能：生产计划、生产调度、材料管理、生产控制、质量管理和库存控制

- **事件(event)，又称事务(transaction)：必须作为一个整体完成的逻辑单位工作。由离散的输入触发，当过程与相应的输出响应时事件结束。**
- **每一个事件都有一个触发器和响应，可以用事件的输入和输出定义。**
- **功能由响应事件的过程组成。每个业务事件由响应那个事件的一个过程表示。采用如下规则命名：“处理__”、“响应__”或“生成__”**

例如：处理客户订单、处理客户地址修改、响应客户投诉、生成退货报告和生成发票

逻辑过程 Logical Processes

事件过程可以进一步分解成详细说明系统必须如何响应事件的基本过程

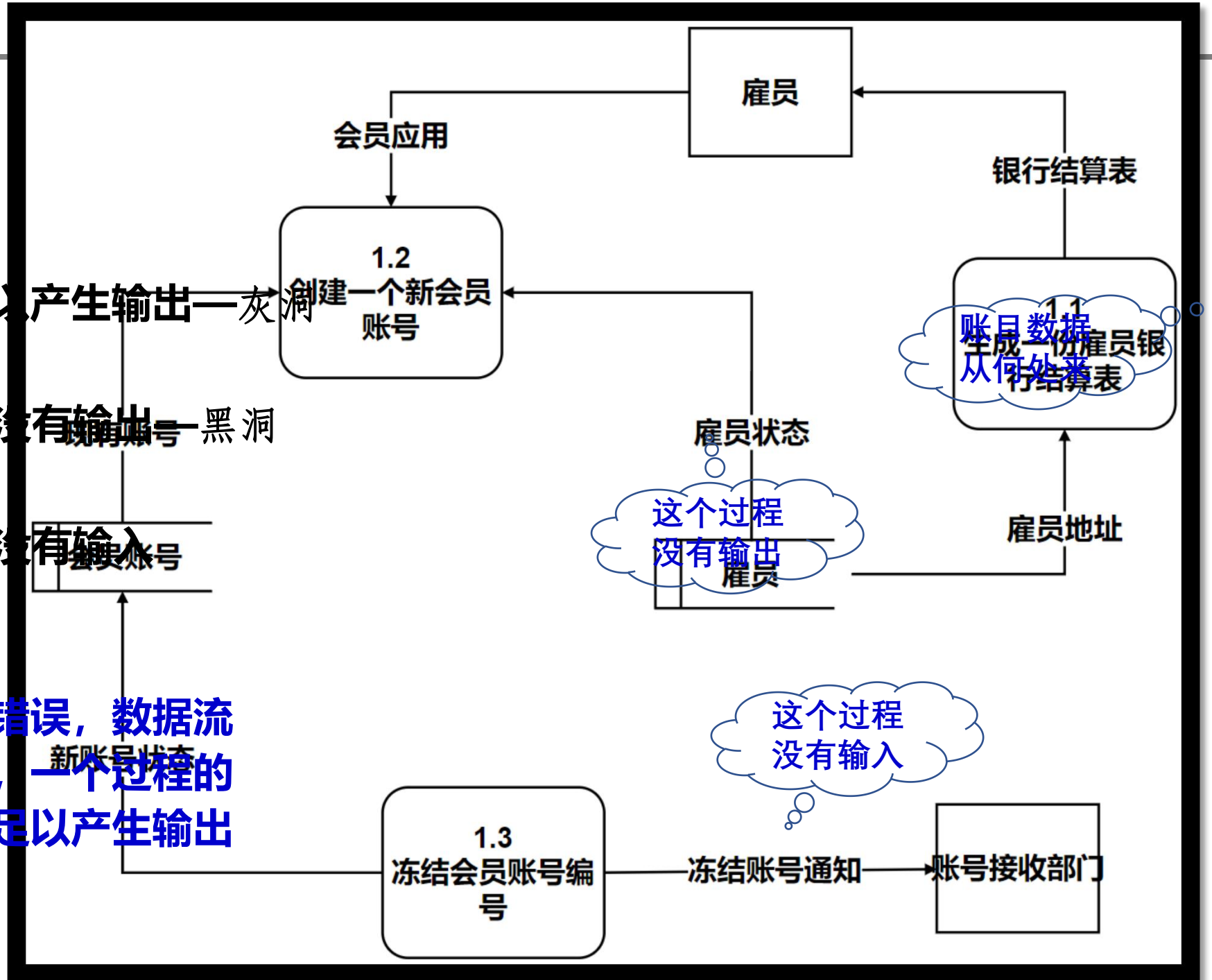
- **基本过程(elementary process)，又称原子过程(primitive process)：**
  - **为完成一个事件的响应所需要的的离散的详细活动或任务**
  - 是在一个过程模型中描述的最低层次细节
  - 用一个强动作动词后跟一个描述实施的工作的宾语从句命名

例如：验证客户身份、检查产品质量、计算订单费用、排序退单、获得客户地址、修改客户地址、增加新客户和删除客户

逻辑过程 Logical Processes

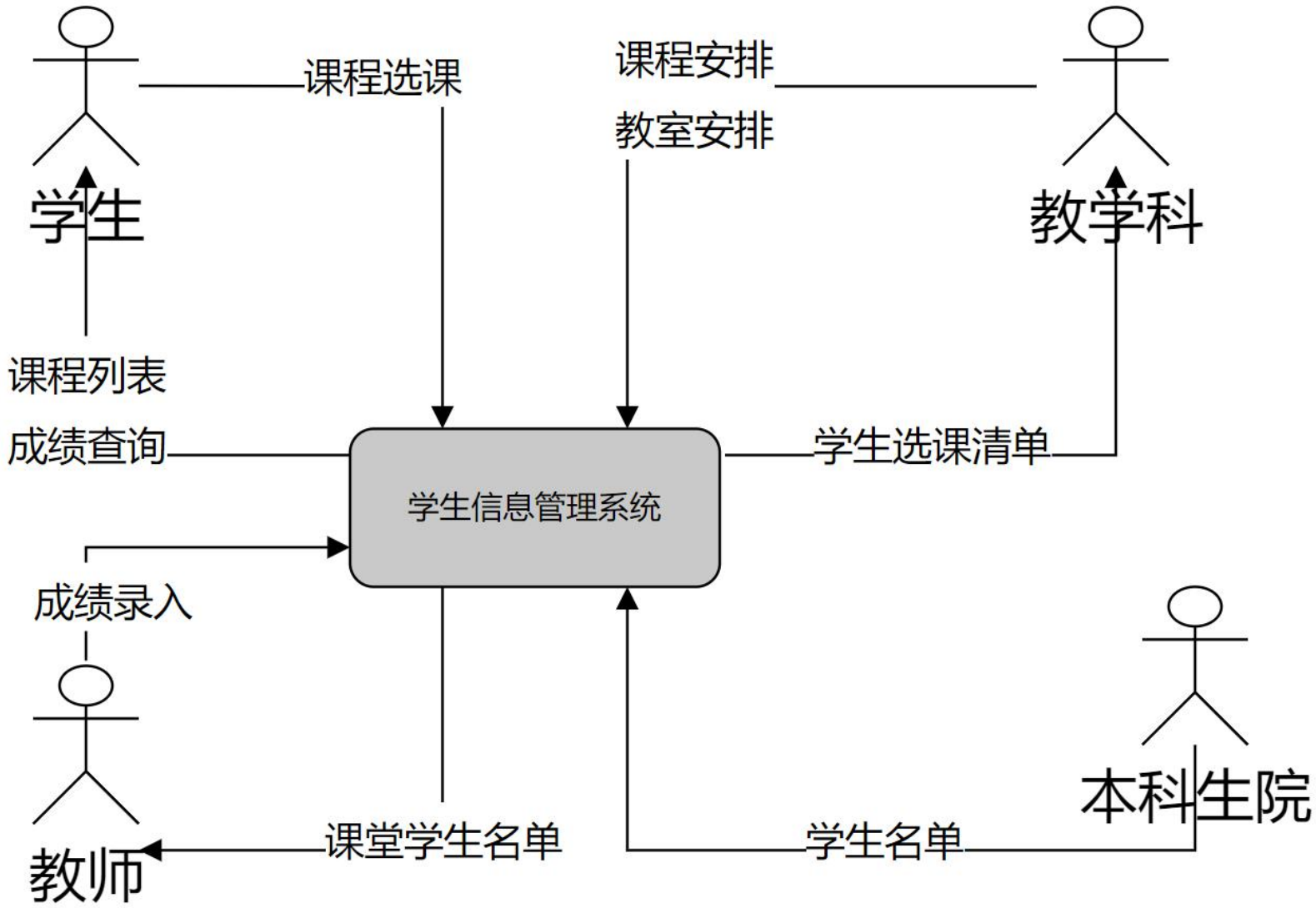
- 忽略移动或路由数据（保持数据不变）而不做任何其他工作的过程
- 仅保留以下逻辑过程：
  - 执行计算 （计算绩点）
  - 做出决策（决定是否可以获得订购的产品）
  - 排序、过滤或总结数据（确定过期的发票）
  - 组织数据成为有用的信息（生成报告或回答问题）
  - 触发其他过程（打开抽屉或指挥机器人）
  - 使用存储的数据（创建、删除、修改或读取记录）

- 1.1 输入不足以产生输出——灰洞 (gray hole)
- 1.2 有输入但没有输出——黑洞 (black hole)
- 1.3 有输出但没有输入

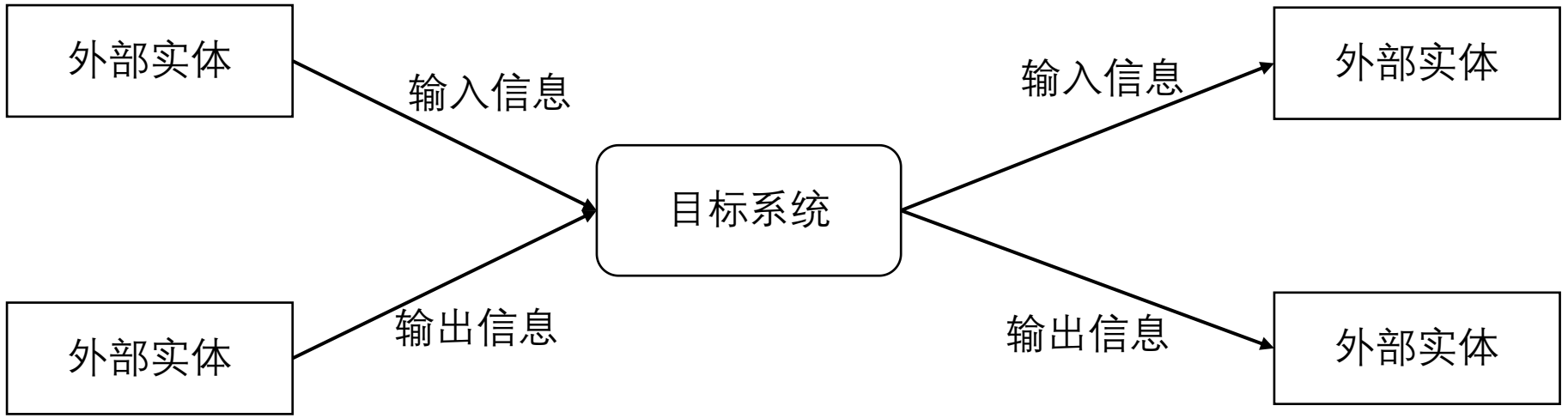


灰洞是最常见的错误，数据流图交给程序员时，一个过程的输入数据流必须足以产生输出数据流

上下文图 Context Diagram



典型的上下文数据流图



练习：机票预定系统DFD

——只需要画上下文数据流图

系统功能：旅行社把预定机票的旅客信息输入机票预定系统；系统为旅客安排航班，打印出取票通知单，返给旅行社；旅客在飞机起飞前凭取票单取票，系统检验无误后，输出机票给旅客。

【腾讯文档】飞机票预定系统DFD
https://docs.qq.com/form/page/DSGR6dm1qU25ab2xi

练习：招生系统DFD

——只需要画上下文数据流图

学校首先公布招生条件，考生根据自己的条件报名，之后系统进行资格审查，并给出资格审查信息；

对于资格审查合格的考生可以参加答卷，系统根据学校提供的试题及答案进行自动判卷，并给出分数及答题信息，供考生查询；

最后系统根据学校的录取分数线进行录取，并将录取信息发送给考生。

【腾讯文档】招生系统DFD
https://docs.qq.com/form/page/DSE1UeXhEU3pIcnlJ

本节内容

Readings

《系统分析与设计方法》

- 第9章 过程建模

- 关键词：结构化方法；过程建模；上下文图；数据流图；事件图；基本图；结构化英语