

## 9 信息系统分析与设计

# 目录

- 9.1 信息系统开发方法概述
- 9.2 系统规划
- 9.3 系统分析
- 9.4 系统设计
- 9.5 系统实施
- 9.6 系统运行和维护



## 9.1 信息系统开发方法概述



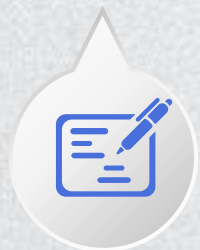
## 系统开发过程管理

将软件工程和系统工程方法引入，

系统的



规范的



定量的



- 系统生命周期法
- 原型法
- 面向对象法
- 计算机辅助软件工程法





## 系统生命周期法

### ➤ 什么是系统生命周期？

一个信息系统从**目标提出**到系统设计、**实现、应用**直到最终**完成系统使命**的全过程。

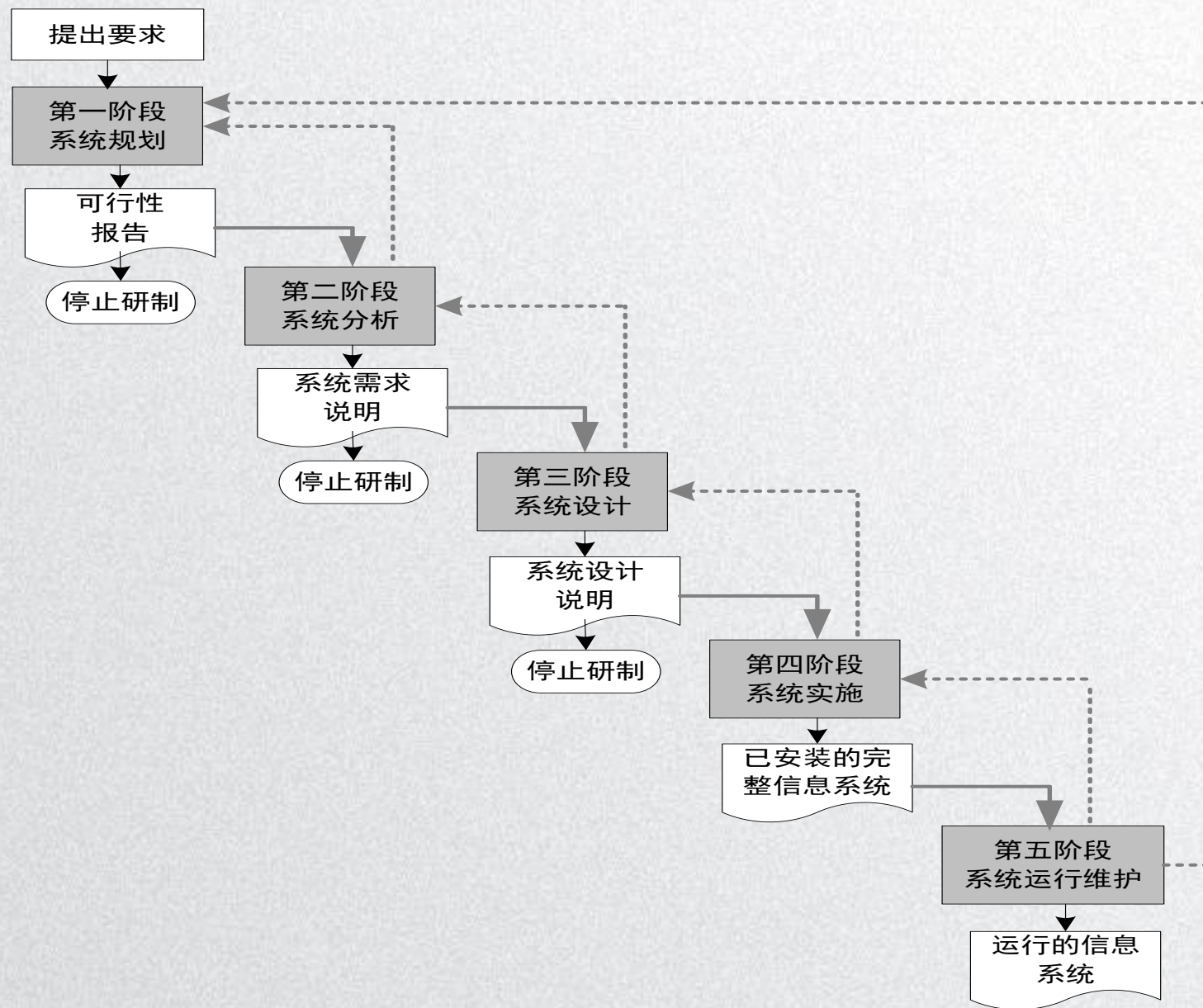
### ➤ 系统生命周期法：

在系统**生命周期全过程**分阶段进行开发管理。

**基本思想**：“自顶向下，逐步求精”

指采用**结构化**分析方法，从全局出发，将复杂的系统逐步分解细化，直到可以实现。

# 生命周期法开发过程的瀑布模型







## 系统生命周期法

### 优点

自顶向下，强调系统开发过程的**整体性和全局性**。

**划分阶段**，降低了复杂性，提高可操作性。

**文档审核**，保证系统质量，提高可维护性。

阶段互相依赖，不能并行，**开发周期长**。

**前期错误向后传递**，修正代价高。

**难以适应需求变化**，不支持反复开发。

### 缺点

分析重于设计，设计重于编码！



## 原型法

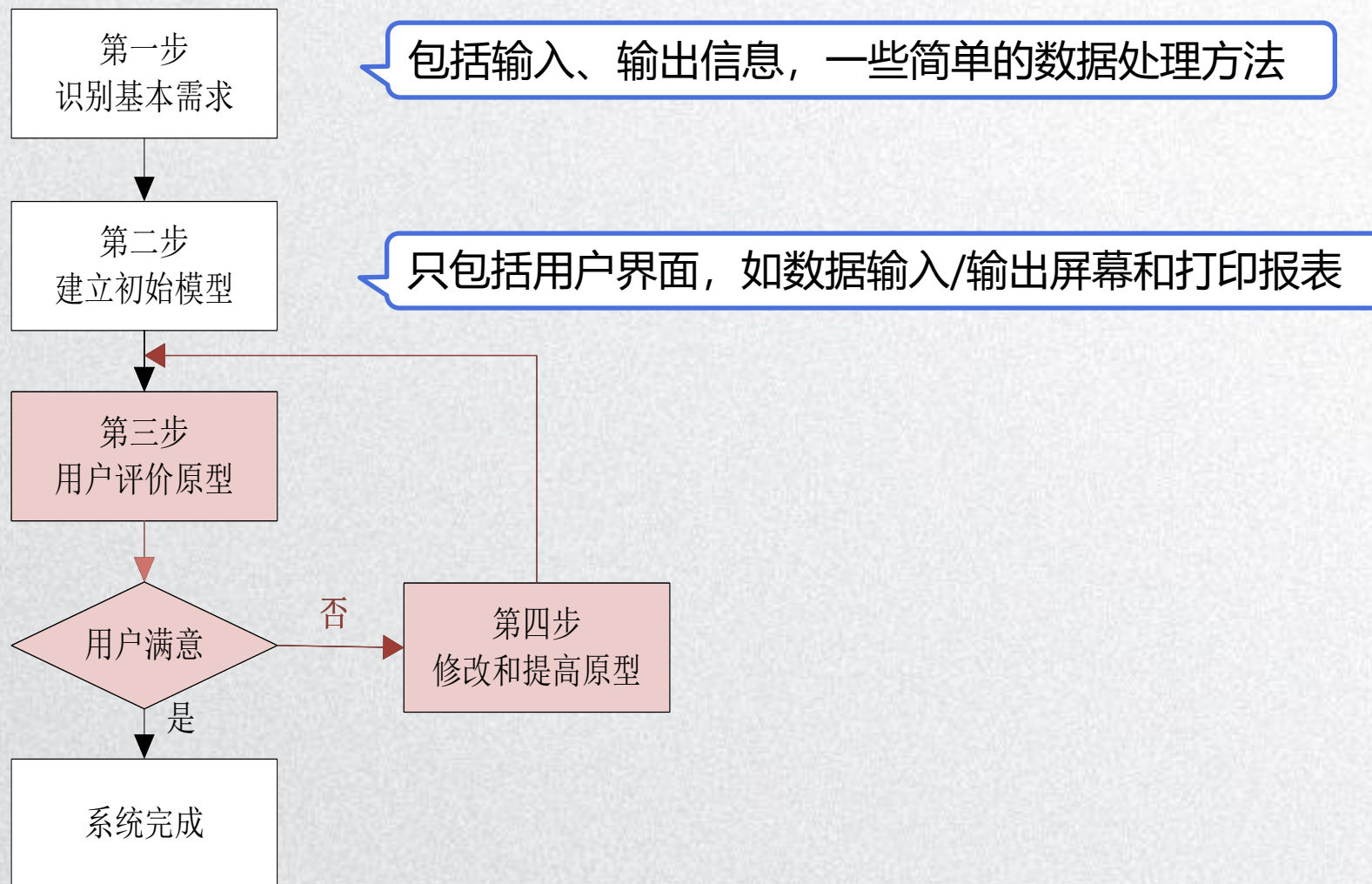
### 什么是原型法？

根据最初获得的用户基本需求，快速建立一个目标系统版本，即原型，然后根据用户评价修改提高原型，最终形成实际系统。

基本思想：“模拟原型，逐步完善”



# 原型法开发过程





## 原型法

### 优点

流程简单，简化分析和设计  
用户与开发者易于沟通，鼓励用户参与  
对于业务简单的小型系统，开发周期短

要准确理解需求，不适合复杂系统  
缺乏统一规划和标准，易导致失控  
易忽略测试和文档，后期维护困难

### 缺点



## 信息系统开发方法

- 系统生命周期法
- 原型法

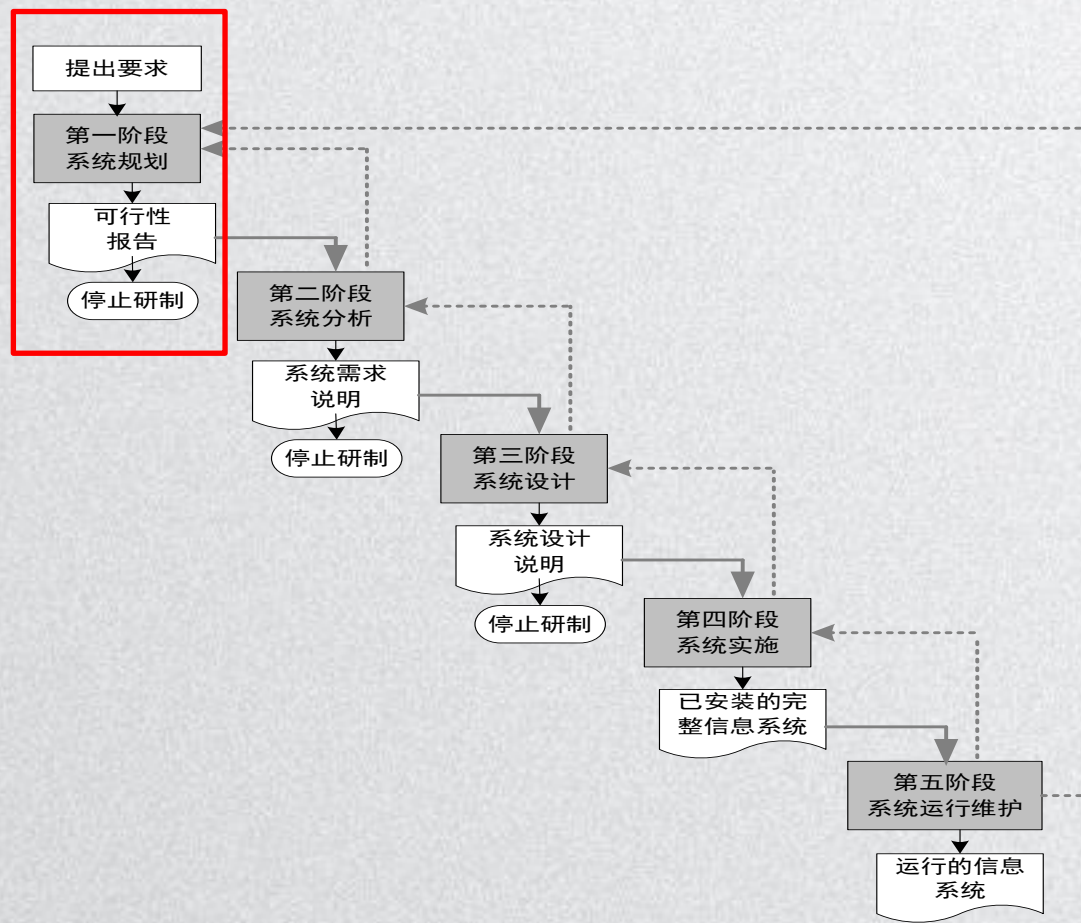
**THANK YOU!**



## 9.2 系统规划



# 系统规划

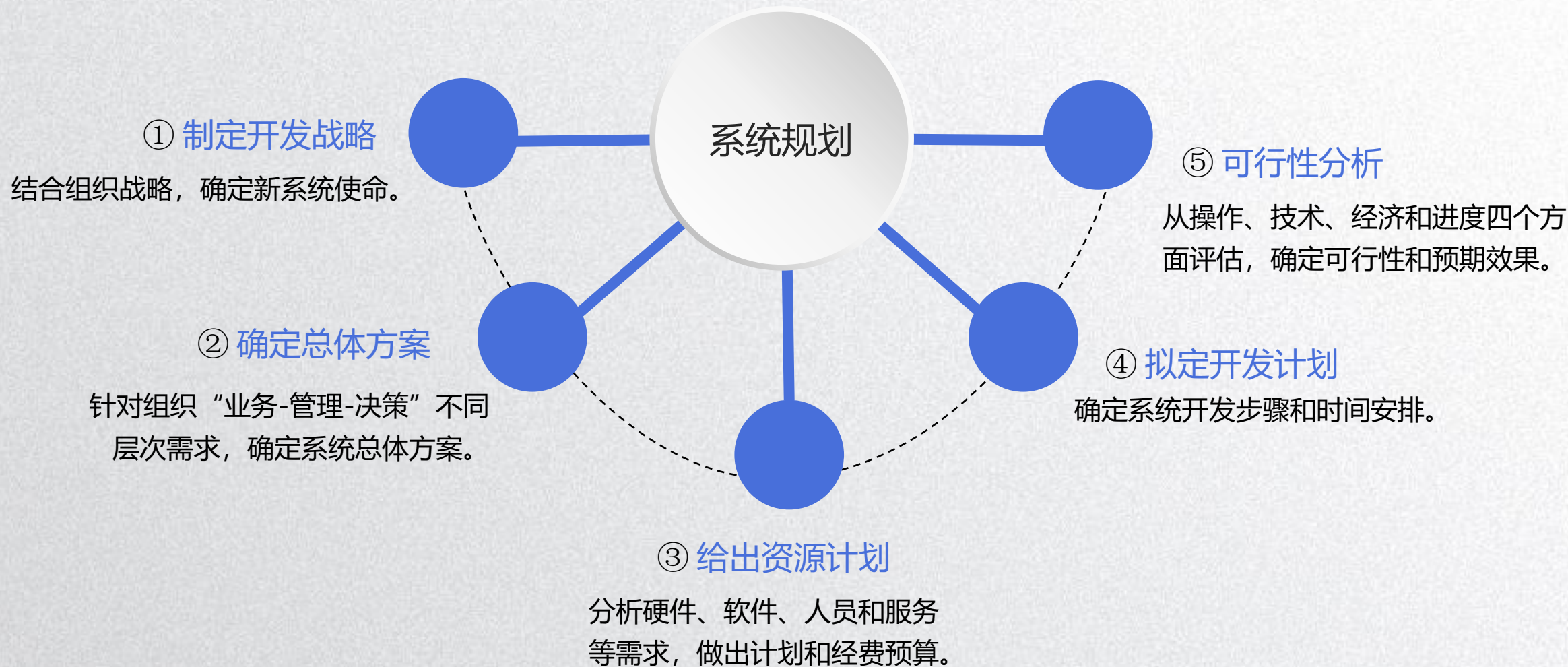


**系统规划**是系统生命周期的第一个阶段，在这一阶段要明确系统目标、规模和开发计划，解决“**为什么做**和**能不能做**”的问题，**完成可行性分析报告**。

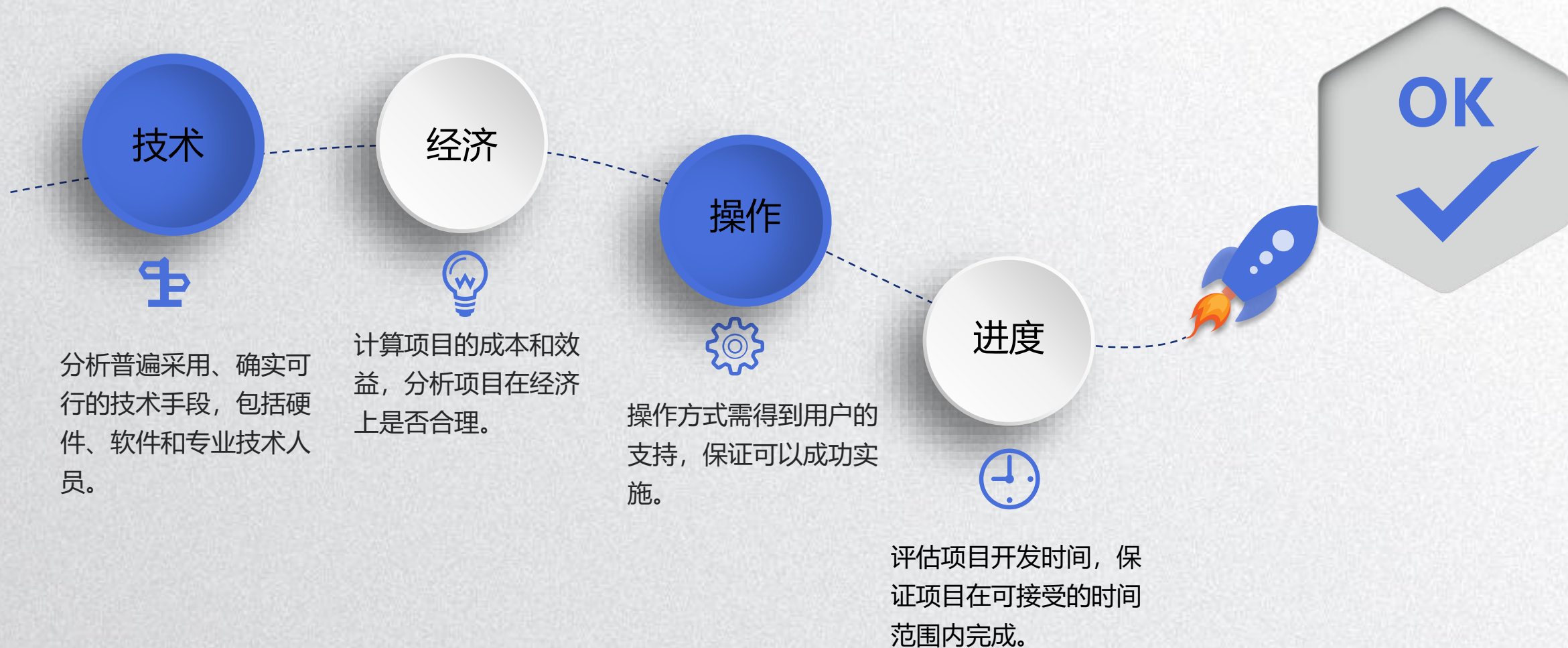




## 系统规划的主要任务



# 可行性分析





## 系统规划

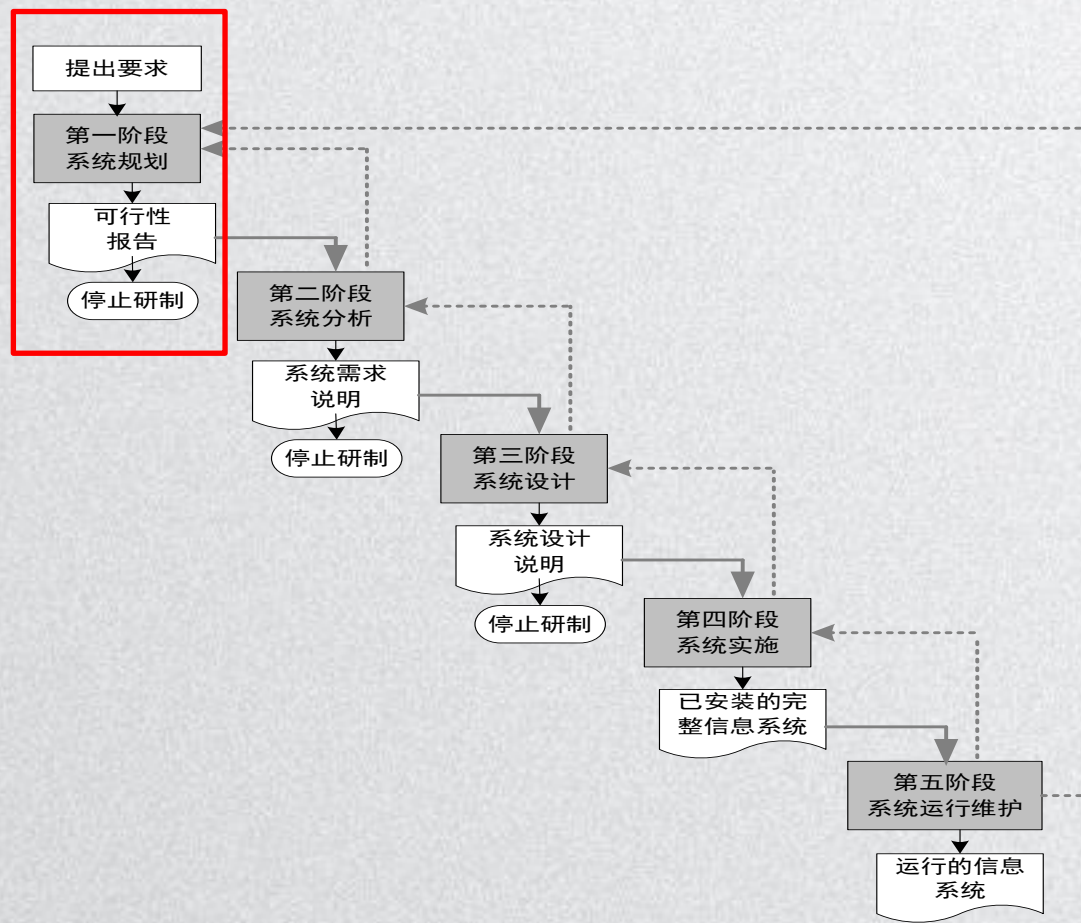


### 可行性分析报告

- ① **引言**。说明系统名称、系统目标、系统功能以及系统的由来。
- ② **系统建设的背景、必要性和意义**。详尽说明系统规划调查和汇总的过程，让人信服调查是真实的，汇总是有根据的，规划是可信的。
- ③ **候选方案**。可提出一个主方案和多个辅助方案。
- ④ **可行性论证**。要从技术、经济、操作、进度等方面的可行性给予充分论证。
- ⑤ **确定方案和拟定开发计划**。包括开发进度以及各阶段人员、资金、设备的需求。



# 系统规划



**系统规划**决定系统的目标，它要求组织的高层领导参与，是后续工作的重要前提。



**THANK YOU!**

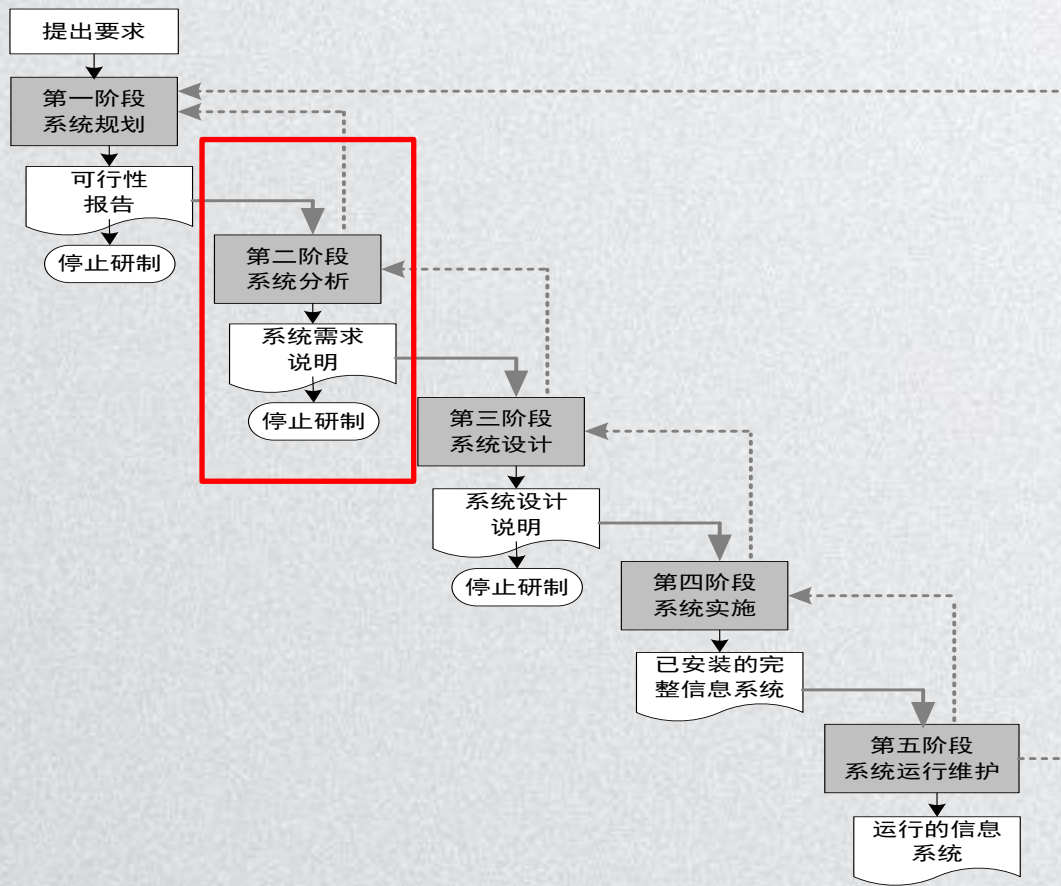
## 9.3 系统分析





# 系统分析

系统分析是系统生命周期的第二个阶段，要进行细致的需求调研和梳理工作，明确系统需要“**做什么**”，形成“**系统需求说明**”报告。





# 系统需求发现

## 如何进行需求发现？

- ✓ 面谈
- ✓ 观察业务操作
- ✓ 查阅文档和报表
- ✓ 问卷调查
- ✓ 抽样和研究



调查清楚用户对信息系统的实际需求，包括信息内容和处理要求。

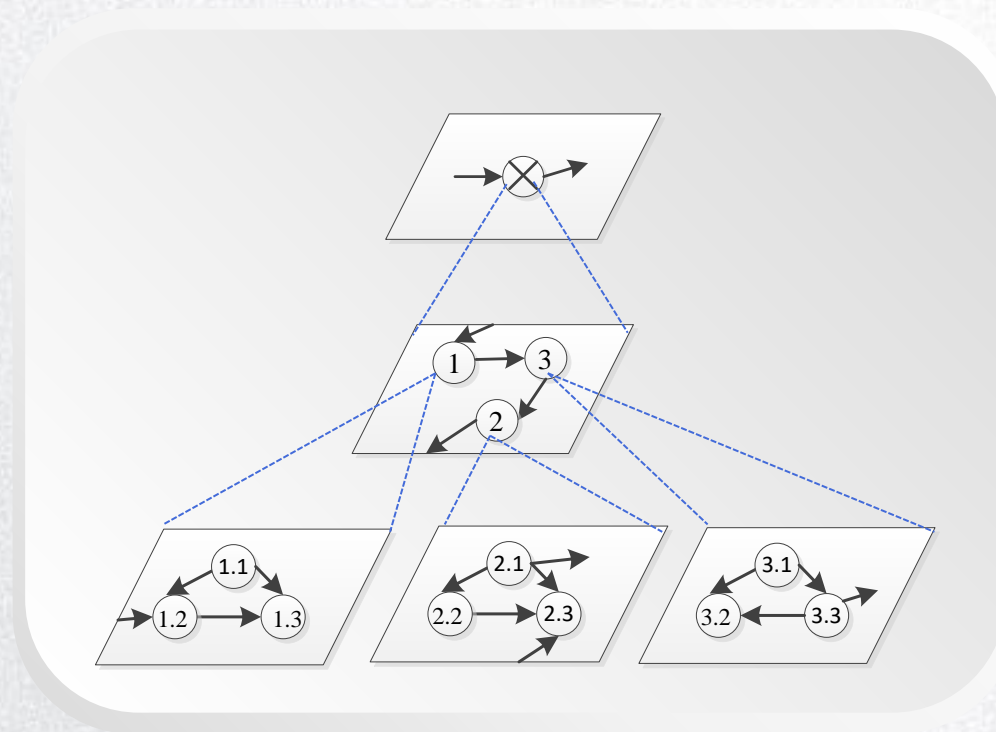




什么是**结构化分析方法**？

Structured Analysis(SA), 是一种**面向数据流**的需求分析方法。它以数据在不同模块中移动的观点来看待一个系统。

**自顶向下，逐层分解。**





## 需求建模技术

采用**结构化**方法，对**数据**和**处理过程**建模来详细描述系统需求，就形成系统的**逻辑模型**。

常用建模工具有：

- ✓ 数据流图
- ✓ 数据字典
- ✓ 决策表或决策树



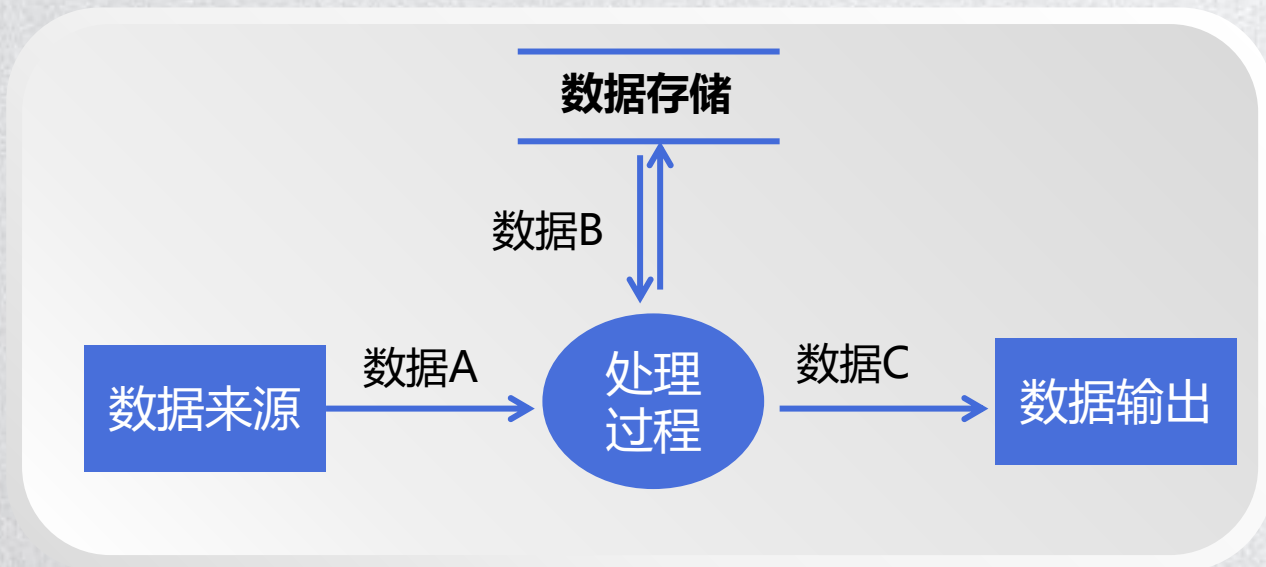
## 数据流图

数据流图 (Data Flow Diagram, DFD)

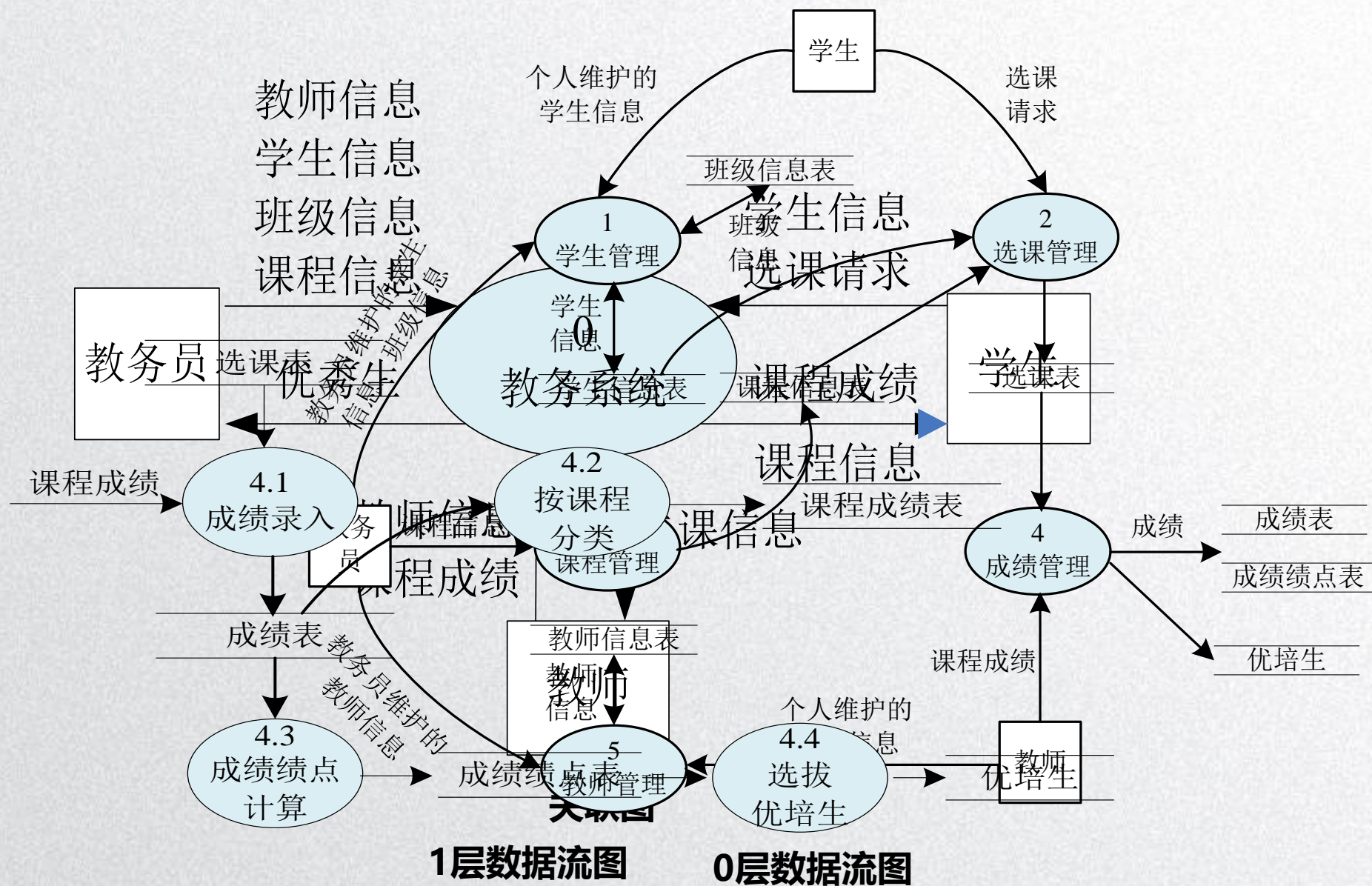
从数据传递和加工的角度来刻画数据流从输入到输出的移动变换过程。数据流图表达了数据和处理过程的关系。

图中，

- 命名的箭头表示数据流，
- 椭圆表示处理过程，
- 矩形表示外部实体，
- 上下平行的双直线表示数据存储。



## 举例：教务系统数据流图





# 数据字典

---

数据字典（Data Dictionary, DD）是关于数据的信息集合，它是对数据流图中包含的所有元素的定义。



# 举例：教务系统数据字典

表1 实体说明

实体名	含义说明	输入数据流	输出数据流
学生	在学校有学籍的学生	课程成绩	学生信息、选课请求
教师	在学校任教有工号的教师	选课信息	教师信息、课程成绩
教务员	在学校有工号的教务管理人员	优培生	教师信息、学生信息、班级信息、课程信息

实体说明描述各实体的含义、输入数据流和输出数据流



表2 数据流说明

数据流	含义说明	数据来源	数据去向	组成
个人维护的学生信息	学生自己可维护的信息	学生	学生信息表	联系电话, 个人介绍, 密码
教务员维护的学生信息	教务员维护的学生信息	教务员	学生信息表	学号, 姓名, 性别, 出生日期, 是否住校, 联系电话, 照片, 个人介绍, 密码, 班号
个人维护的教师信息	教师自己可维护的信息	教师	教师信息表	联系电话, 个人介绍, 密码
教务员维护的教师信息	教务员维护的教师信息	教务员	教师信息表	工号, 姓名, 学院, 性别, 入校时间, 职称, 联系电话, 照片, 个人介绍, 密码, 是否教务员
班级信息	一个班级的信息	教务员	班级信息表	班号, 班级名, 专业名, 班长
选课请求	一个学生选一门课	学生、 课程信息表	选课表	学号, 课程号
课程信息	一门课程的基本信息	教务员	课程信息表	课程号, 课程名, 学时数, 学分, 开设时间, 任课教师
课程成绩	一个学生选一门课的成绩	教师	成绩表	学号, 课程号, 成绩, 录入时间
成绩	学生某门课程的成绩	成绩管理	成绩表 成绩绩点表	学号, 姓名, 课程号, 课程名, 教师号, 成绩, 绩点

数据流说明描述各数据流的内容、来源和去向

表3 数据记录说明

记录名	含义说明	数据项组成
学生	一个学生的有关信息	学号，姓名，性别，出生日期，班号，是否住校，联系电话，照片，个人介绍，密码
班级	一个班级的有关信息	班号，班级名，专业名，班长
教师	一个教师的有关信息	工号，姓名，学院，性别，入校时间，职称，联系电话，照片，个人介绍，密码，是否教务员
课程	一门课程的有关信息	课程号，课程名，学时数，学分，开设学期，任课教师，上课时间，上课地点，上课学时，实验学时，最大学生数，成绩分析
选课	一个学生选一门课的有关信息	学号，课程号，成绩，录入时间
优培生	一个优培生的有关信息	学号，认定时间，学分绩点

数据记录说明描述各处理数据记录的含义和数据项组成



表4 数据项说明（学生实体）

数据项	含义说明	类型	长度	取值范围	取值含义	与其他数据项关系	其他说明
学生实体							
学号	唯一标识一个学生	字符型	4	0101~9999	前2位表示班级号，后2位按顺序编号	主关键字	
姓名	学生姓名	字符型	<=20	中英文均可，允许重名		不能为空	
性别	学生性别	字符型	1	“男”或“女”		不能为空	
出生日期	学生生日	日期型					
是否住校	住校还是走读	逻辑型		0或1	1表示住校		默认为1
联系电话	座机、手机等	字符型	<=40				多个电话以“，”分隔
照片	二寸标准照	字符型	<=50		照片存放的路径和文件名		
个人介绍		文本型	0~10000		个人的一段文字介绍		
密码	登录密码	文本型	<=16				
班号	班级编号	字符型	2		所属的班级编号	空或班级记录的存在值	

数据项说明描述各数据项的详细含义和约束等

表5 处理过程说明（学生管理、教师管理.....）

**处理过程名：学生管理**

输入数据：学生信息、班级信息

输出数据：学生信息表、班级信息表

处理过程：

- ①对学生信息进行查询和维护（包括增加、修改和删除）；学生密码的修改。
- ②对班级信息进行查询和维护（包括增加、修改和删除）；按班级或专业生成学生名单。

**处理过程名：教师管理**

输入数据：教师信息

输出数据：教师信息表

处理过程：

对教师信息进行查询和维护（包括增加、修改和删除教师）；教师密码的修改。

**处理过程名：选课管理**

输入数据：选课请求、学生信息表、课程信息表

输出数据：选课表

处理过程：

①选课：查询课程信息及授课教师信息，选择多门课程，提交选择课程的编号，检测所选课程上课时间以及与已有课程是否有时间冲突，如果不冲突则一个学生选择的多门课程记录新增至选课表，并且被选课程的剩余名额减1，否则提示冲突回到选课页面。

②退课：查询选课表，选择退课课程号，从选课表删除课程，并使该课程剩余名额加1。

③生成选课学生名单报表（点名册）。

.....

**处理过程说明**详细描述各处理的输入、输出和功能。



# 判定表和判定树

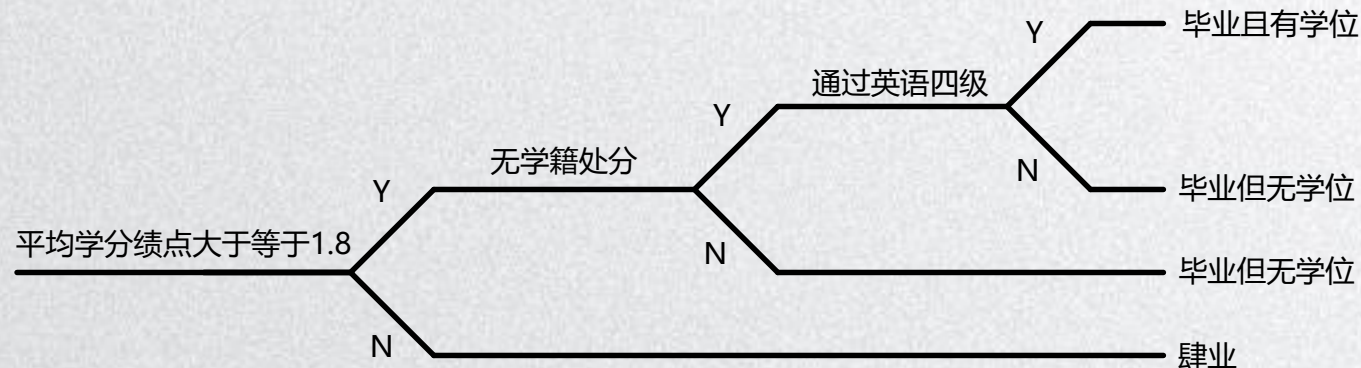
判定表（判定树）用表或树结构清晰地描述多逻辑条件下的处理行为。

例如：教务系统学生毕业处理的条件和处理行为表达

判定表

		1	2	3	4	5	6	7	8
条件	平均学分绩点大于等于1.8	Y	Y	Y	Y	N	N	N	N
	无学籍处分	Y	Y	N	N	Y	Y	N	N
	通过英语四级	Y	N	Y	N	Y	N	Y	N
动作	毕业且有学位	X							
	毕业但无学位		X	X	X				
	肄业					X	X	X	X

判定树



## 系统 分析



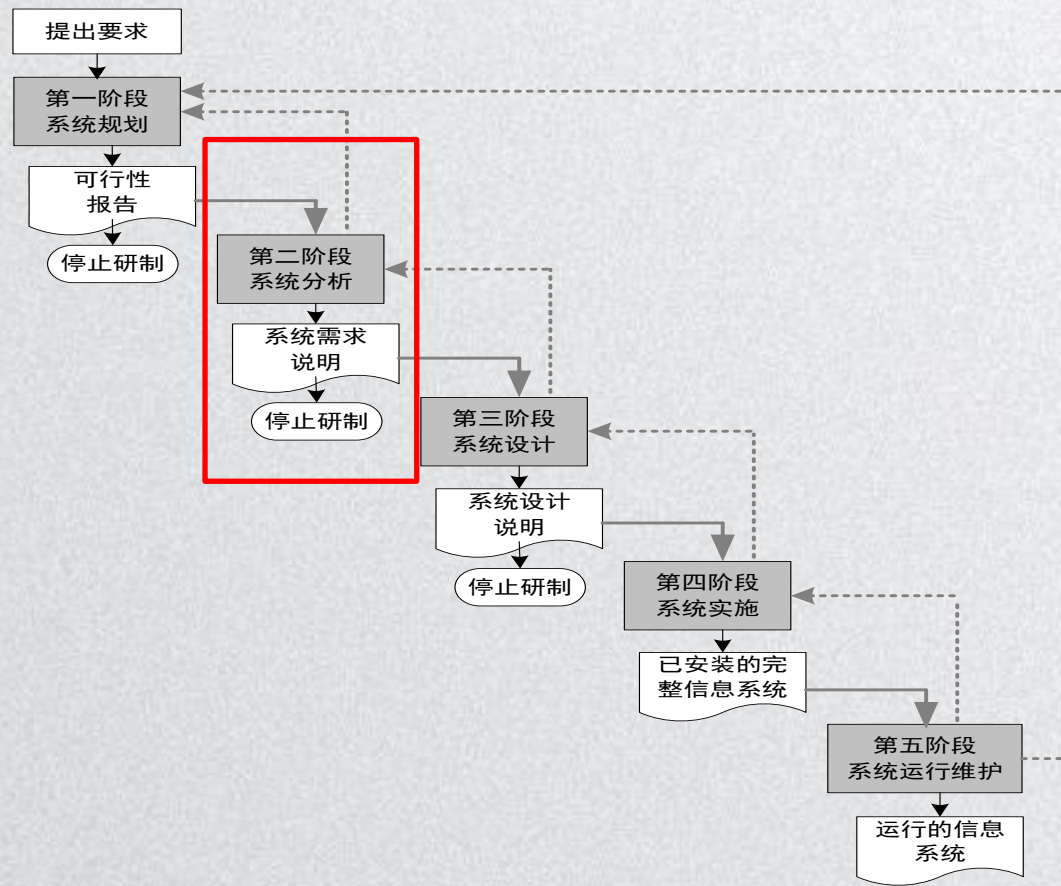
### 需求分析说明

- ① **引言。**说明编写目的、系统名称、背景、任务提出者及相关者；给出专门术语定义；列出参考资料。
- ② **任务概述。**叙述系统开发意图、应用目标、作用范围及其他有关背景材料；说明用户类别和特点；列出系统开发工作的假定和约束条件。
- ③ **需求规定。**对系统功能的规定，描述数据字典、各功能的输入、输出及处理过程；对系统性能的规定等。
- ④ **运行环境规定。**硬件、软件、外部通信接口等。
- ⑤ **待确定问题的列表。**





# 系统分析



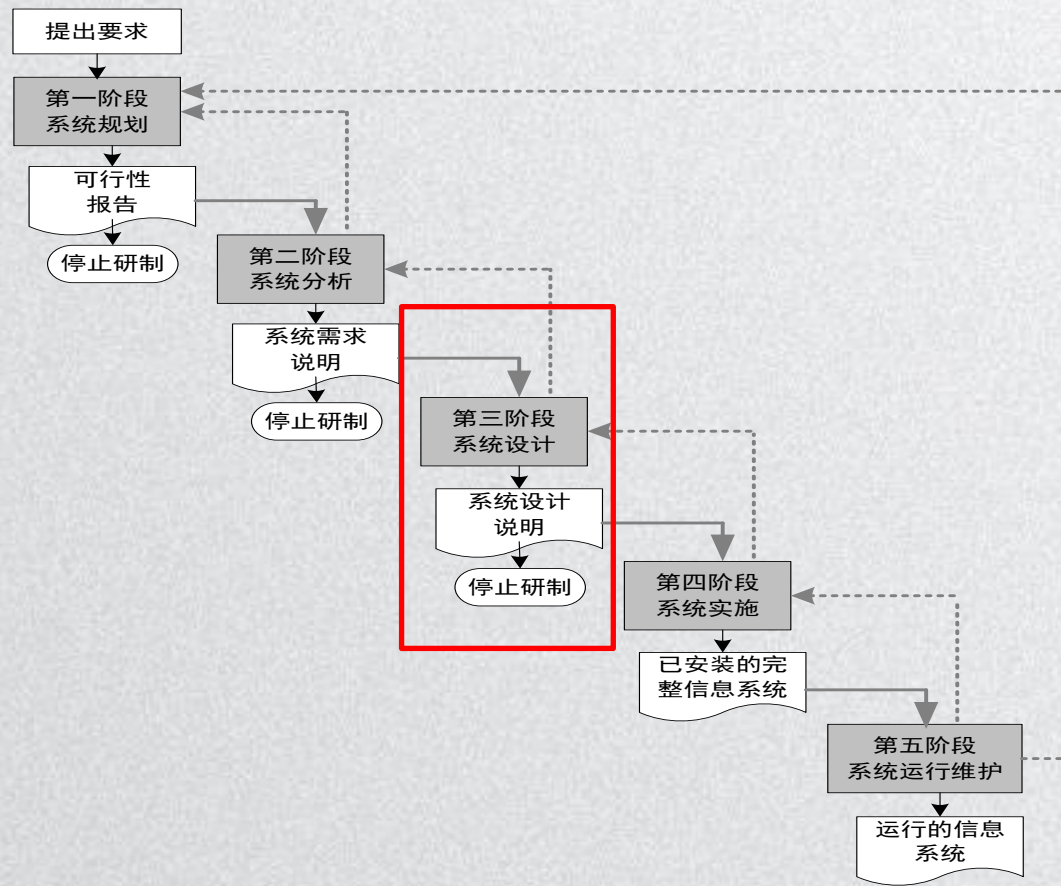
**THANK YOU!**



## 9.4 系统设计概述



# 系统设计



在系统分析的基础上，给出实现系统目标的完整方案，需要解决“**怎么做？**”的问题。





# 系统设计



# 系统设计



## 系统概要设计说明书

- ① 系统架构
- ② 功能结构
- ③ 数据库
- ④ 页面总体风格
- ⑤ 出错处理



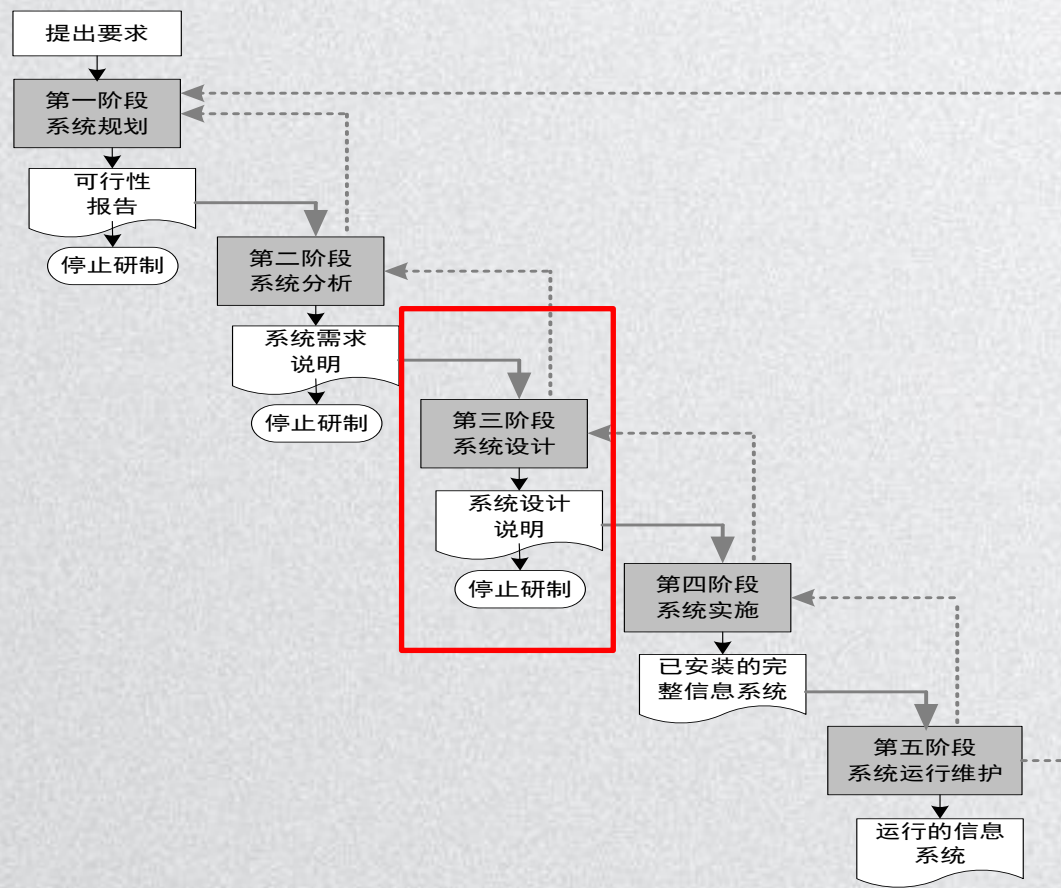
## 系统详细设计说明书

对每个功能模块的 每个窗口或页面的布局、  
处理流程、实现算法、数据输入/输出等。





# 系统设计



**THANK YOU!**

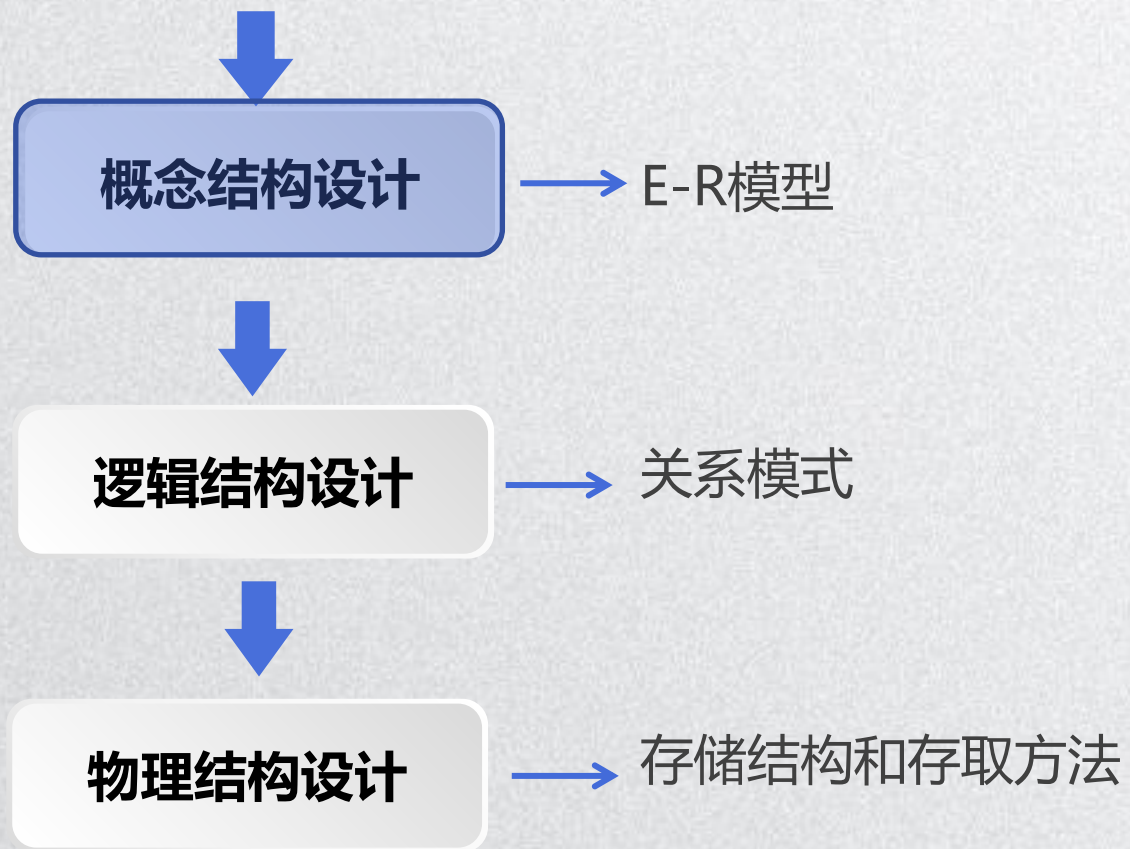


## 9.4 系统设计

### ——数据库概念结构设计



# 数据库设计

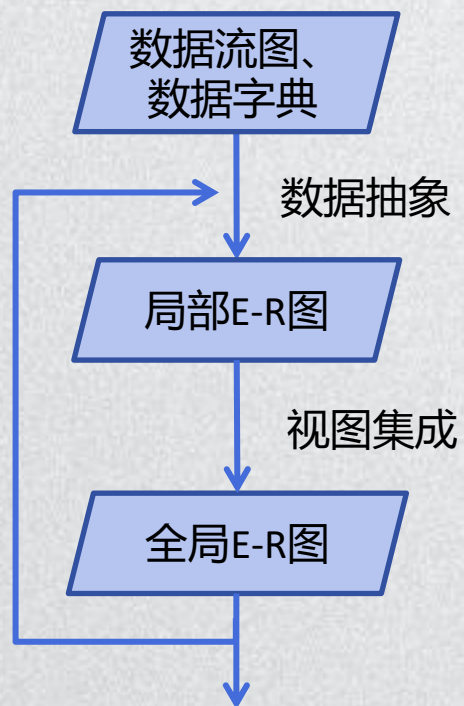






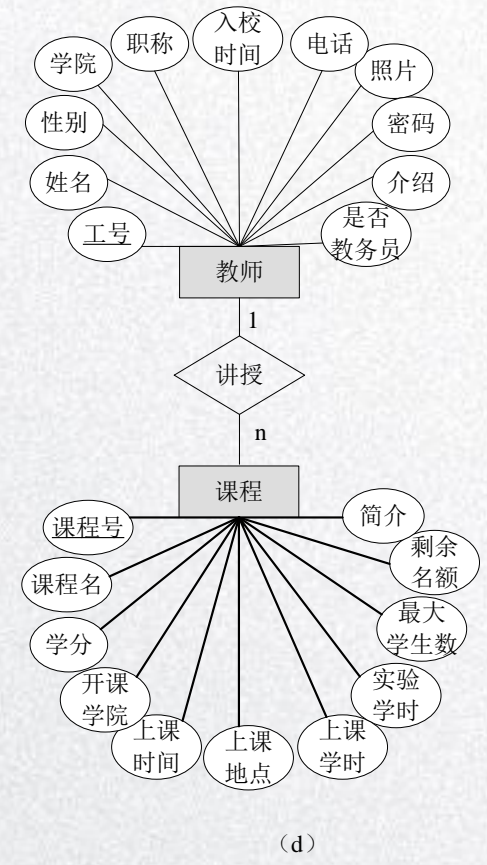
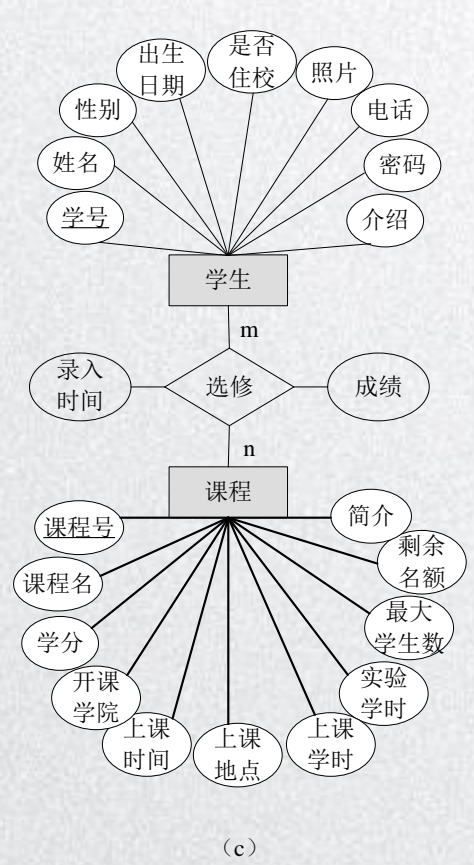
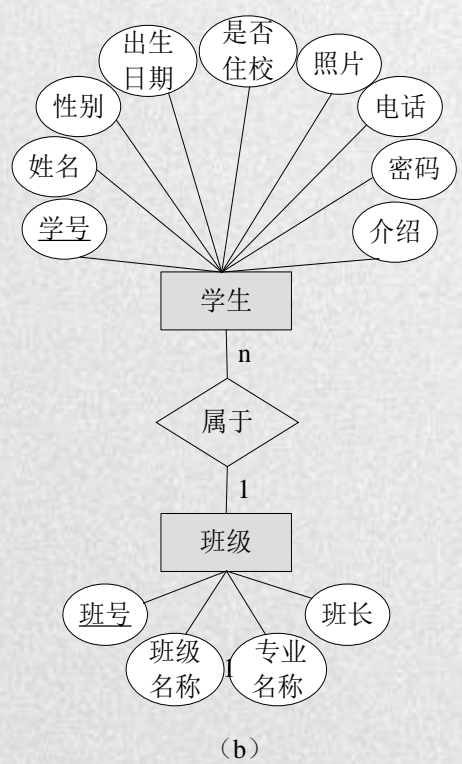
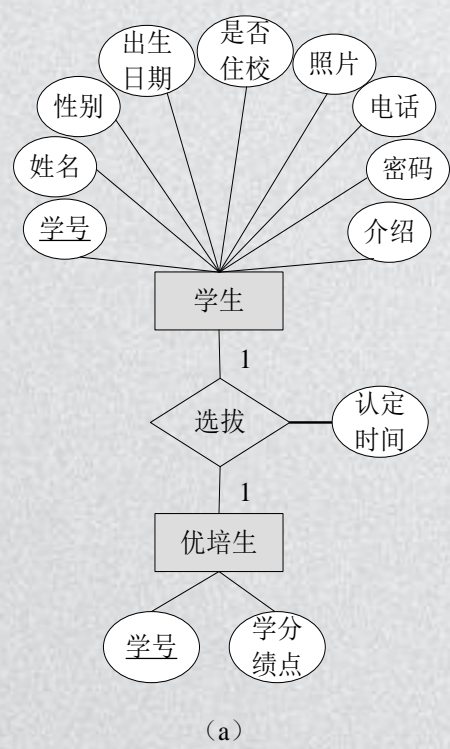
## 数据库概念结构设计

将系统分析得到的信息处理需求抽象为概念结构，主要描述工具是E-R模型。



# 数据库概念结构设计——局部E-R图

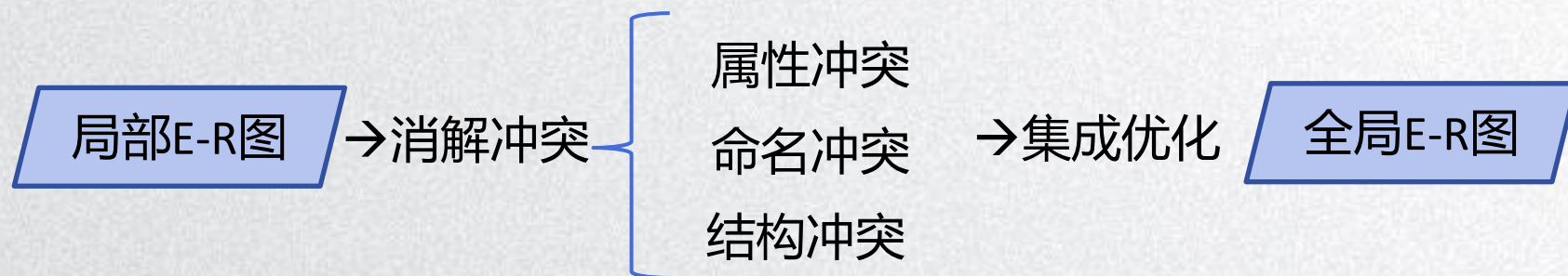
实体 → 实体间的联系 → 实体属性和联系属性 → 实体的主关键字





# 数据库概念结构设计——全局E-R图

---



学生 (学号【字符串】, 姓名, 性别, 生日)  
优培生 (学号【整数】, 年龄)  
选课 (学生代号, 课号, 成绩)  
成绩 (数学, 外语, 计算机)

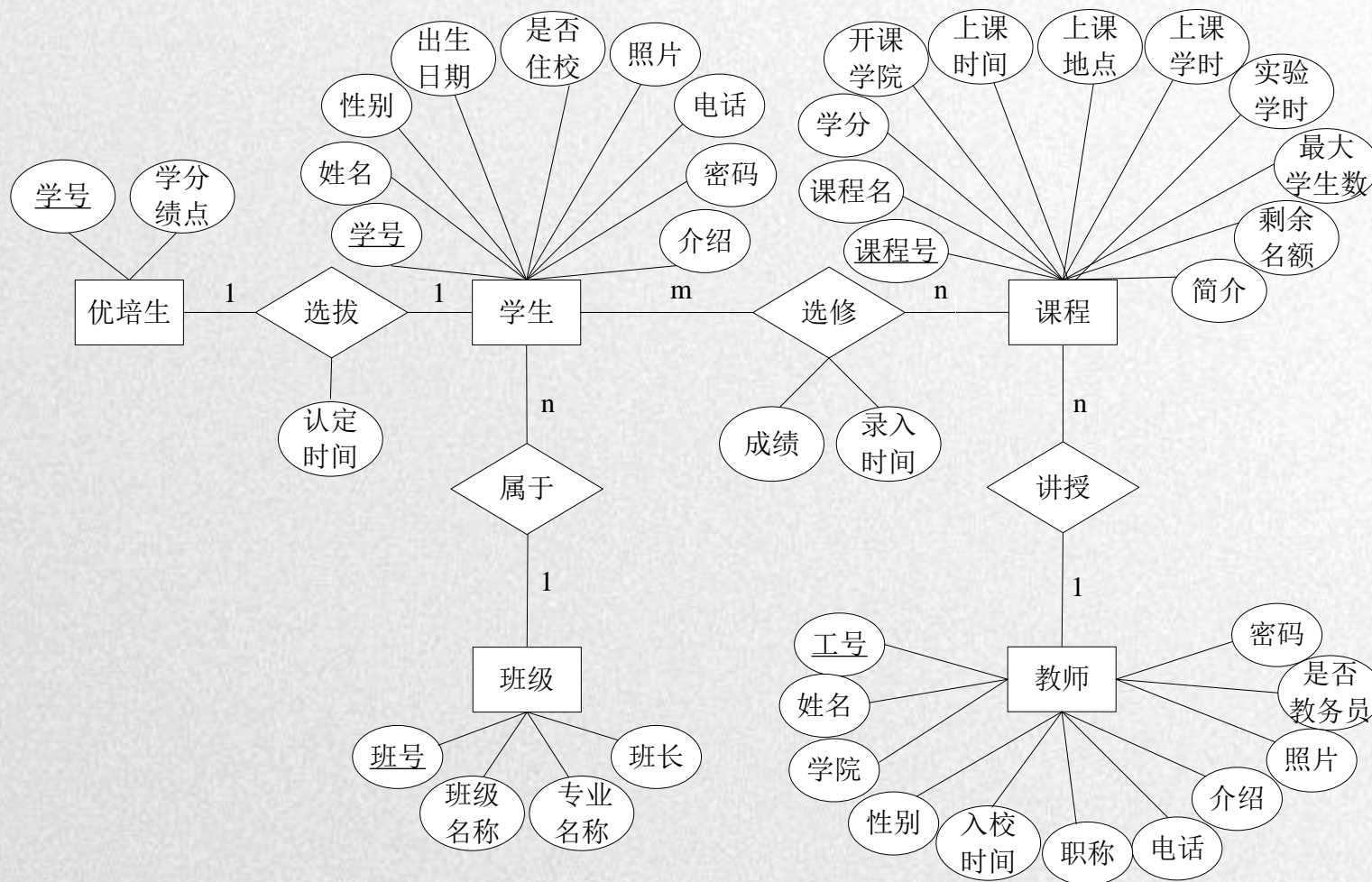
# 数据库概念结构设计——全局E-R图

反应  
需求

实体少

属性少

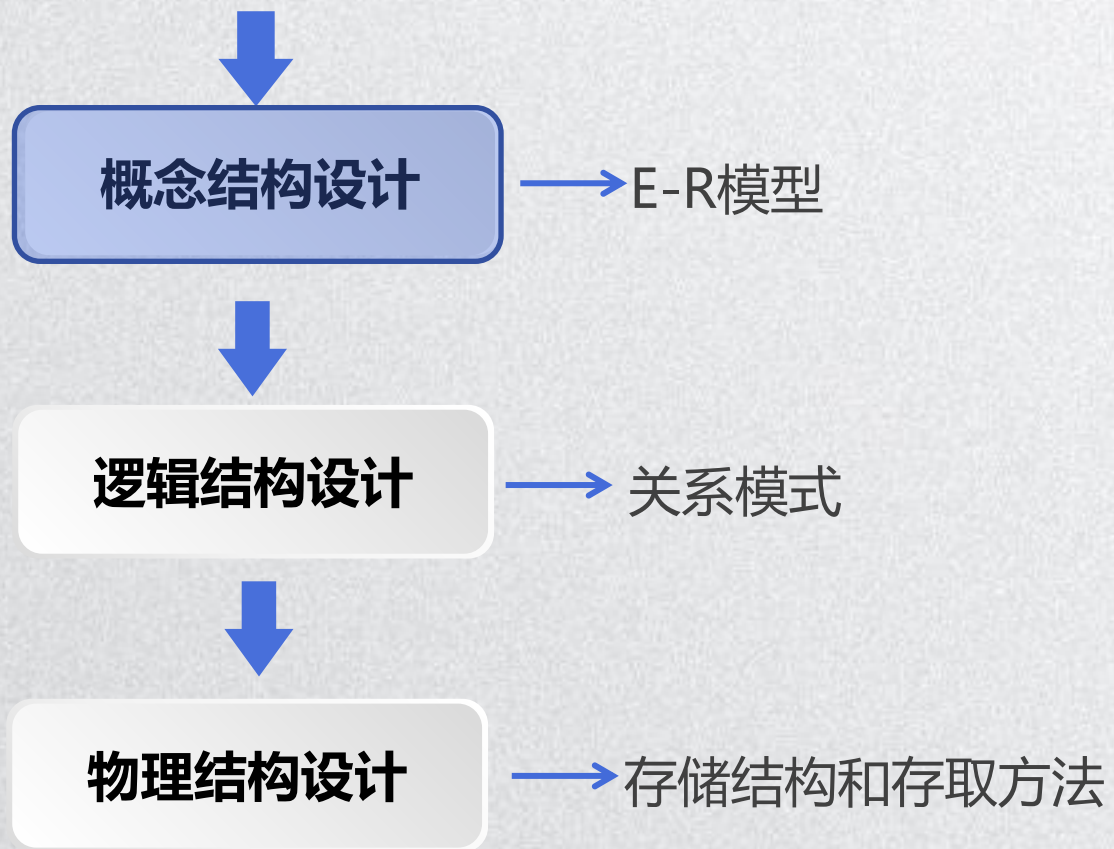
联系  
无冗余







## 小结：数据库设计



**THANK YOU!**

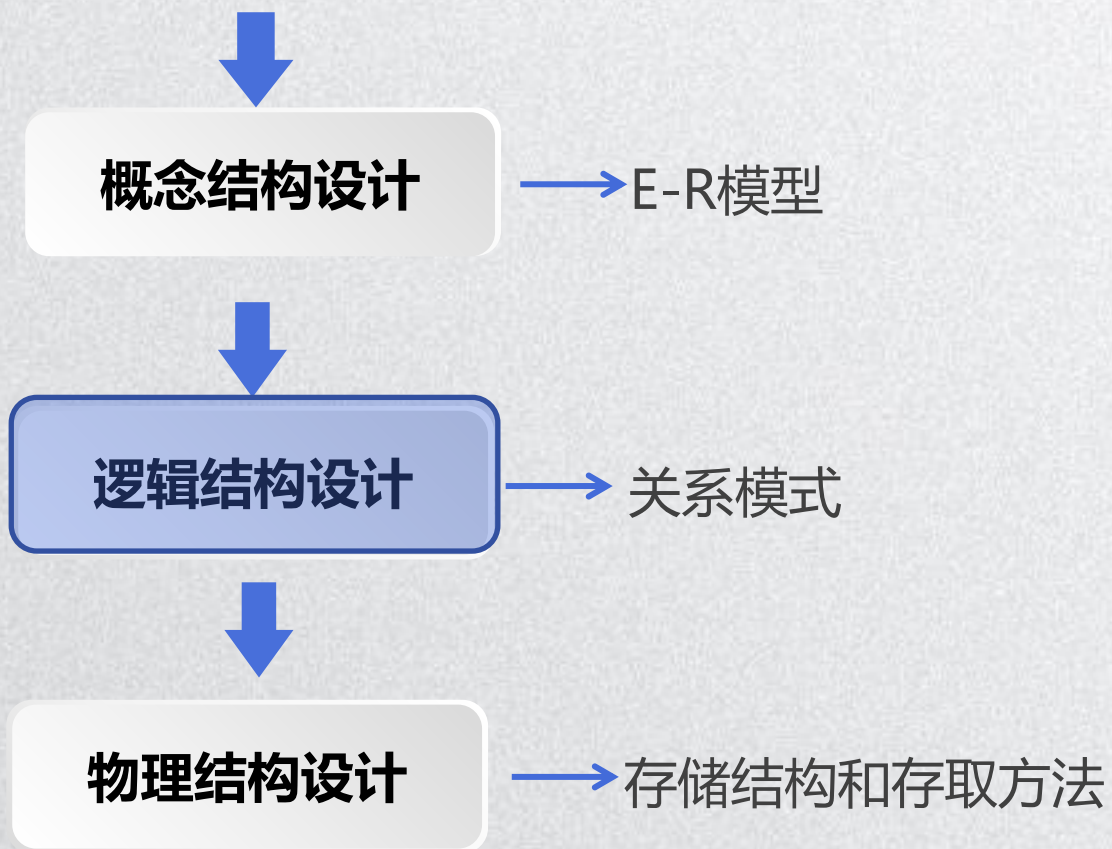


## 9.4 系统设计

### ——数据库逻辑结构设计



# 数据库设计

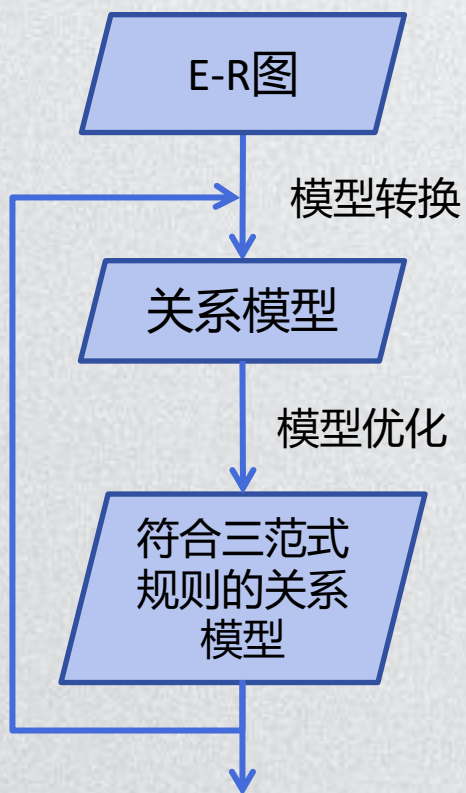






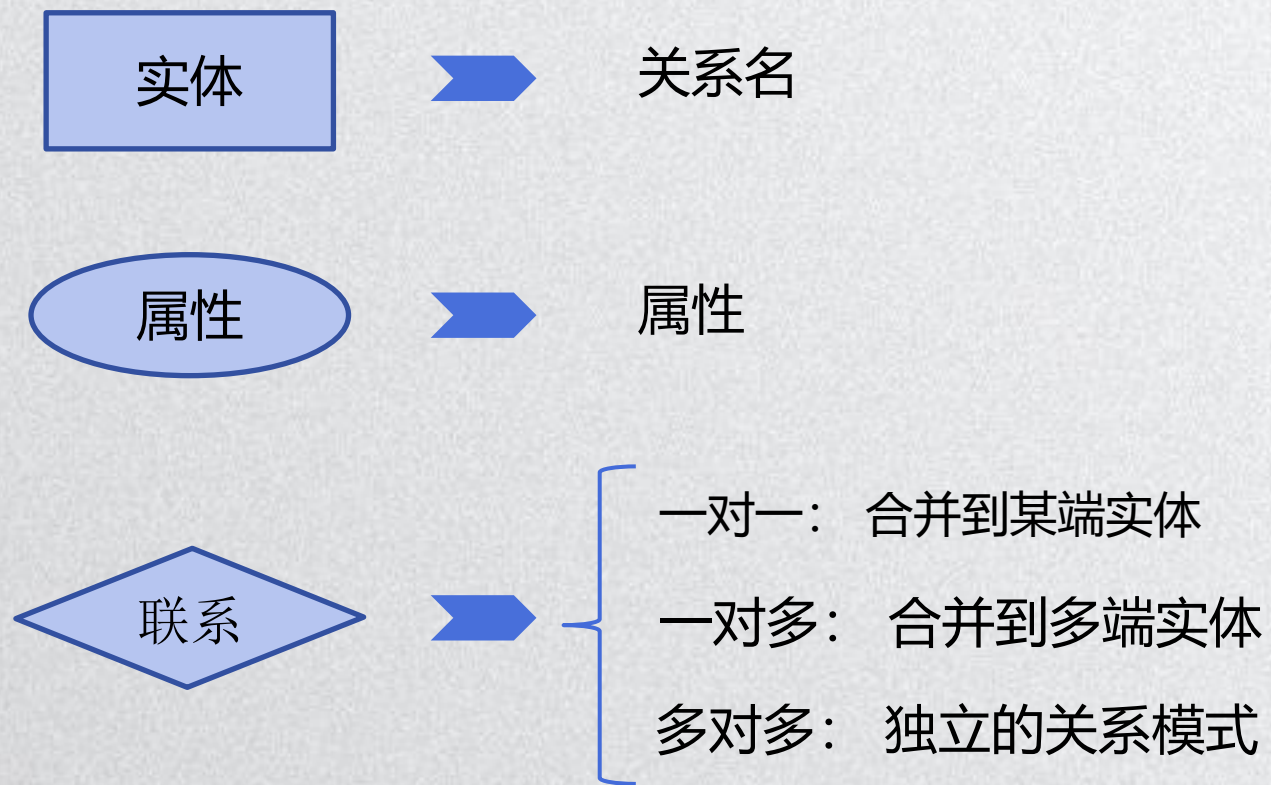
## 数据库逻辑结构设计

把概念结构E-R模型转换为关系数据模型并进行优化，就形成关系数据库逻辑结构。



# 数据库逻辑结构设计——E-R模型转换关系模型

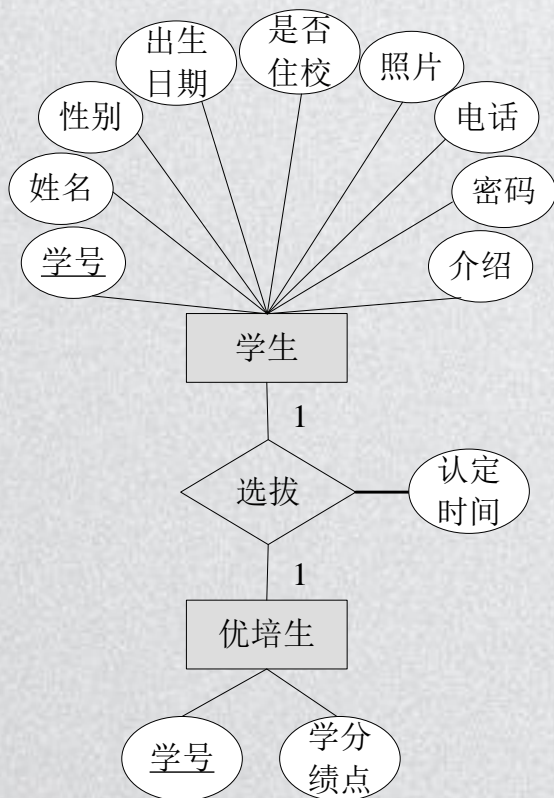
---





# 数据库逻辑结构设计——E-R模型转换关系模型

- ✓ 一对一联系：将联系的属性和任意“1端”的主键合并到另一端的关系模式中。



## 方案1

学生（学号，姓名，性别，出生日期，是否住校，照片，电话，密码，介绍）

优培生（学号，学分绩点，**认定时间**）

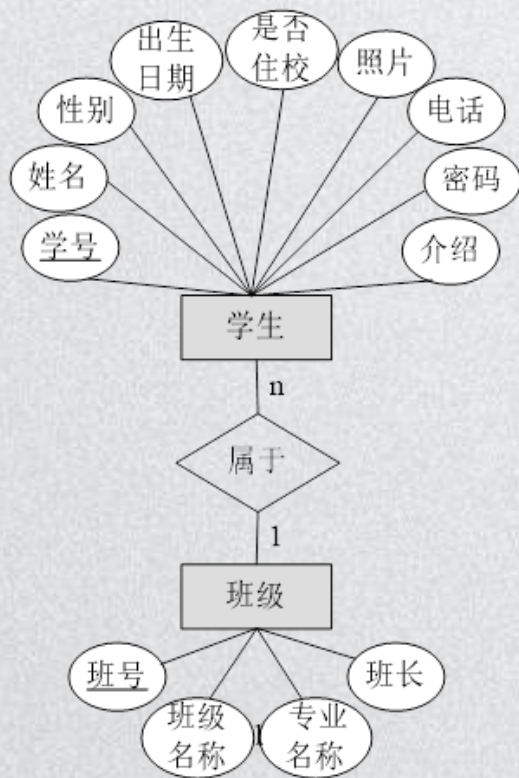
## 方案2

学生（学号，姓名，性别，出生日期，是否住校，照片，电话，密码，介绍，**认定时间**）

优培生（学号，学分绩点）

# 数据库逻辑结构设计——E-R模型转换关系模型

- ✓ 一对多联系：将联系的属性和“1端”的主键合并到“多端”的关系模式中。



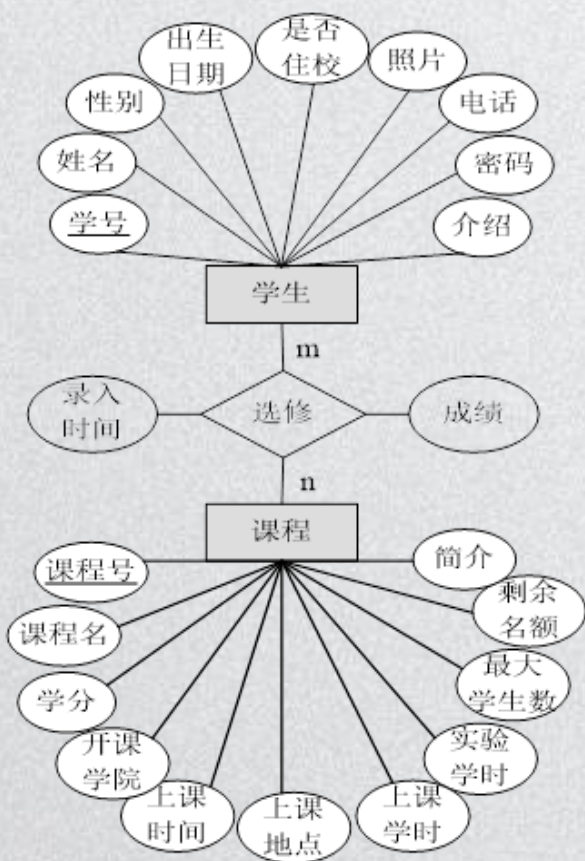
学生（学号，姓名，性别，出生日期，是否住校，照片，电话，密码，介绍，**班号**）

班级（班号，班级名称，专业名称，班长）



# 数据库逻辑结构设计——E-R模型转换关系模型

- ✓ 多对多联系：将联系的属性及两端的主键合并到一个新的关系模式中。



学生（学号，姓名，性别，出生日期，是否住校，照片，电话，密码，介绍，认定时间）

课程（课程号，课程名，学分，开课学院，上课时间，上课地点，上课学时，实验学时，最大学生数，剩余名额，简介）

选修（学号，课程号，成绩，录入时间）

# 数据库逻辑结构设计——关系模型优化

## 关系模型优化的目标？

- ✓ 减少数据冗余
- ✓ 避免操作异常
- ✓ 避免数据不一致

**第一范式 (1NF)**

**第二范式 (1NF)**

**第三范式 (1NF)**

关系规范化理论



# 数据库逻辑结构设计——关系模型优化

---

## ① 第一范式

关系模式中的每一分量都不可分

## ② 第二范式

关系模式中的所有非主键属性都完全依赖于主关键字

## ③ 第三范式

关系模式中的任意非主键属性都不传递依赖于主关键字

## 举例：关系模型规范化——1NF

学号	姓名		课程名	学分	开课教师	教师职称	成绩
	姓	名					
1001	陈	佳迪	高等数学	4	张晓芸	讲师	90
			大学物理	3	马建军	教授	80
			英语	4	李明	副教授	85
1002	张	君悦	高等数学	4	张晓芸	讲师	78
			英语	4	李明	副教授	86
1003	王	宵	数学分析	3.5	张晓芸	讲师	78
1004	林	豆豆	高等数学	4	张晓芸	讲师	45
			哲学	2	王志民	副教授	60

R'

规范化为  
1NF

列属性合并

学号	姓名	课程名	学分	开课教师	教师职称	成绩
1001	陈佳迪	高等数学	4	张晓芸	讲师	90
1001	陈佳迪	大学物理	3	马建军	教授	80
1001	陈佳迪	英语	4	李明	副教授	85
1002	张君悦	高等数学	4	张晓芸	讲师	78
1002	张君悦	英语	4	李明	副教授	86
1003	王霄	数学分析	3.5	张晓芸	讲师	78
1004	林豆豆	高等数学	4	张晓芸	讲师	45
1004	林豆豆	哲学	2	王志民	副教授	60

R

主关键字

第一范式：每一分量都不可分



## 关系模型规范化举例——2NF

学号	姓名	课程名	学分	开课教师	教师职称	成绩
1001	陈佳迪	高等数学	4	张晓芸	讲师	90
1001	陈佳迪	大学物理	3	马建军	教授	80
1001	陈佳迪	英语	4	李明	副教授	85
1002	张君悦	高等数学	4	张晓芸	讲师	78
1002	张君悦	英语	4	李明	副教授	86
1003	王霄	数学分析	3.5	张晓芸	讲师	78
1004	林豆豆	高等数学	4	张晓芸	讲师	45
1004	林豆豆	哲学	2	王志民	副教授	60

R

第二范式：所有非主键属性都完全依赖于主关键字

# 转化为2NF的投影分解法

R (学号, 姓名, 课程名, 学分, 开课教师, 教师职称, 成绩)



用主键属性集的每一个子集作主键构成一个关系模式

R1 (学号, .....)

R2 (课程名, .....)

R3 (学号, 课程名, .....)



将依赖于最小主键的属性放在相应的关系模式中

R1 (学号, 姓名)

R2 (课程名, 学分, 开课教师, 教师职称)

R3 (学号, 课程名, 成绩)



去掉只由主键的子集构成的关系模式

R1 (学号, 姓名)

R2 (课程名, 学分, 开课教师, 教师职称)

R3 (学号, 课程名, 成绩)



# 关系模型规范化举例——2NF

第二范式：所有非主键属性都完全依赖于主关键字

学号	姓名	课程名	学分	开课教师	教师职称	成绩
1001	陈佳迪	高等数学	4	张晓芸	讲师	90
1001	陈佳迪	大学物理	3	马建军	教授	80
1001	陈佳迪	英语	4	李明	副教授	85
1002	张君悦	高等数学	4	张晓芸	讲师	78
1002	张君悦	英语	4	李明	副教授	86
1003	王霄	数学分析	3.5	张晓芸	讲师	78
1004	林豆豆	高等数学	4	张晓芸	讲师	45
1004	林豆豆	哲学	2	王志民	副教授	60

规范化为  
2NF

课程名	学分	开课教师	教师职称
高等数学	4	张晓芸	讲师
大学物理	3	马建军	教授
英语	4	李明	副教授
高等数学	4	张晓芸	讲师
英语	4	李明	副教授
数学分析	3.5	张晓芸	讲师
高等数学	4	张晓芸	讲师
哲学	2	王志民	副教授

R2

学号	课程名	成绩
1001	高等数学	90
1001	大学物理	80
1001	英语	85
1002	高等数学	78
1002	英语	86
1003	数学分析	78
1004	高等数学	45
1004	哲学	60

R3

学号	姓名
1001	陈佳迪
1002	张君悦
1003	王霄
1004	林豆豆

R1

## 关系模型规范化举例——3NF

课程名	学分	开课教师	教师职称
高等数学	4	张晓芸	讲师
大学物理	3	马建军	教授
英语	4	李明	副教授
高等数学	4	张晓芸	讲师
英语	4	李明	副教授
数学分析	3.5	张晓芸	讲师
高等数学	4	张晓芸	讲师
哲学	2	王志民	副教授

R2

学号	课程名	成绩
1001	高等数学	90
1001	大学物理	80
1001	英语	85
1002	高等数学	78
1002	英语	86
1003	数学分析	78
1004	高等数学	45
1004	哲学	60

R3

学号	姓名
1001	陈佳迪
1002	张君悦
1003	王霄
1004	林豆豆

R1

第三范式：任意非主键属性  
都不传递依赖于主关键字



## 转化为3NF的投影分解法

R2 (课程名, 学分, 开课教师, 教师职称)



删除传递依赖于主键的属性

R21 (课程名, 学分, 开课教师)



新建关系模式，放入被依赖者和其所有依赖者，并设被依赖者为主键

R21 (课程名, 学分, 开课教师)

R22 (开课教师, 教师职称)

# 关系模型规范化举例——3NF

课程名	学分	开课教师
高等数学	4	张晓芸
大学物理	3	马建军
英语	4	李明
数学分析	3.5	张晓芸
哲学	2	王志民

R21

开课教师	教师职称
张晓芸	讲师
马建军	教授
李明	副教授
张晓芸	讲师
王志民	副教授

R22

规范化为  
3NF



课程名	学分	开课教师	教师职称
高等数学	4	张晓芸	讲师
大学物理	3	马建军	教授
英语	4	李明	副教授
高等数学	4	张晓芸	讲师
英语	4	李明	副教授
数学分析	3.5	张晓芸	讲师
高等数学	4	张晓芸	讲师
哲学	2	王志民	副教授

R2



第三范式：任意非主键属性  
都不传递依赖于主关键字



学号	姓名		课程名	学分	开课教师	教师职称	成绩
	姓	名					
1001	陈	佳迪	高等数学	4	张晓芸	讲师	90
			大学物理	3	马建军	教授	80
			英语	4	李明	副教授	85
1002	张	君悦	高等数学	4	张晓芸	讲师	78
			英语	4	李明	副教授	86
1003	王	霄	数学分析	3.5	张晓芸	讲师	78
1004	林	豆豆	高等数学	4	张晓芸	讲师	45
			哲学	2	王志民	副教授	60

R'

学号	姓名	课程名	学分	开课教师	教师职称	成绩
1001	陈佳迪	高等数学	4	张晓芸	讲师	90
1001	陈佳迪	大学物理	3	马建军	教授	80
1001	陈佳迪	英语	4	李明	副教授	85
1002	张君悦	高等数学	4	张晓芸	讲师	78
1002	张君悦	英语	4	李明	副教授	86
1003	王霄	数学分析	3.5	张晓芸	讲师	78
1004	林豆豆	高等数学	4	张晓芸	讲师	45
1004	林豆豆	哲学	2	王志民	副教授	60

规范化为  
1NF

课程名	学分	开课教师
高等数学	4	张晓芸
大学物理	3	马建军
英语	4	李明
数学分析	3.5	张晓芸
哲学	2	王志民

R21 (课程表)

开课教师	教师职称
张晓芸	讲师
马建军	教授
李明	副教授
张晓芸	讲师
王志民	副教授

R22 (教师表)

课程名	学分	开课教师	教师职称
高等数学	4	张晓芸	讲师
大学物理	3	马建军	教授
英语	4	李明	副教授
高等数学	4	张晓芸	讲师
英语	4	李明	副教授
数学分析	3.5	张晓芸	讲师
高等数学	4	张晓芸	讲师
哲学	2	王志民	副教授

R2

规范化为  
3NF

学号	姓名
1001	陈佳迪
1002	张君悦
1003	王霄
1004	林豆豆

R1 (学生表)

学号	课程名	成绩
1001	高等数学	90
1001	大学物理	80
1001	英语	85
1002	高等数学	78
1002	英语	86
1003	数学分析	78
1004	高等数学	45
1004	哲学	60

R3 (选课表)

规范化为  
2NF

R' 分解为R1、R21、R22、R3

## 举例：教务系统数据库逻辑结构

---

学生（学号，姓名，性别，出生日期，是否住校，联系电话，照片，个人介绍，密码，班号）

优培生（学号，认定时间，学分绩点）

班级（班号，班级名称，专业名称，班长）

课程（课程号，课程名，学分，开课学院，上课时间，上课地点，上课学时，实验学时，最大学生数，剩余名额，教师工号，简介）

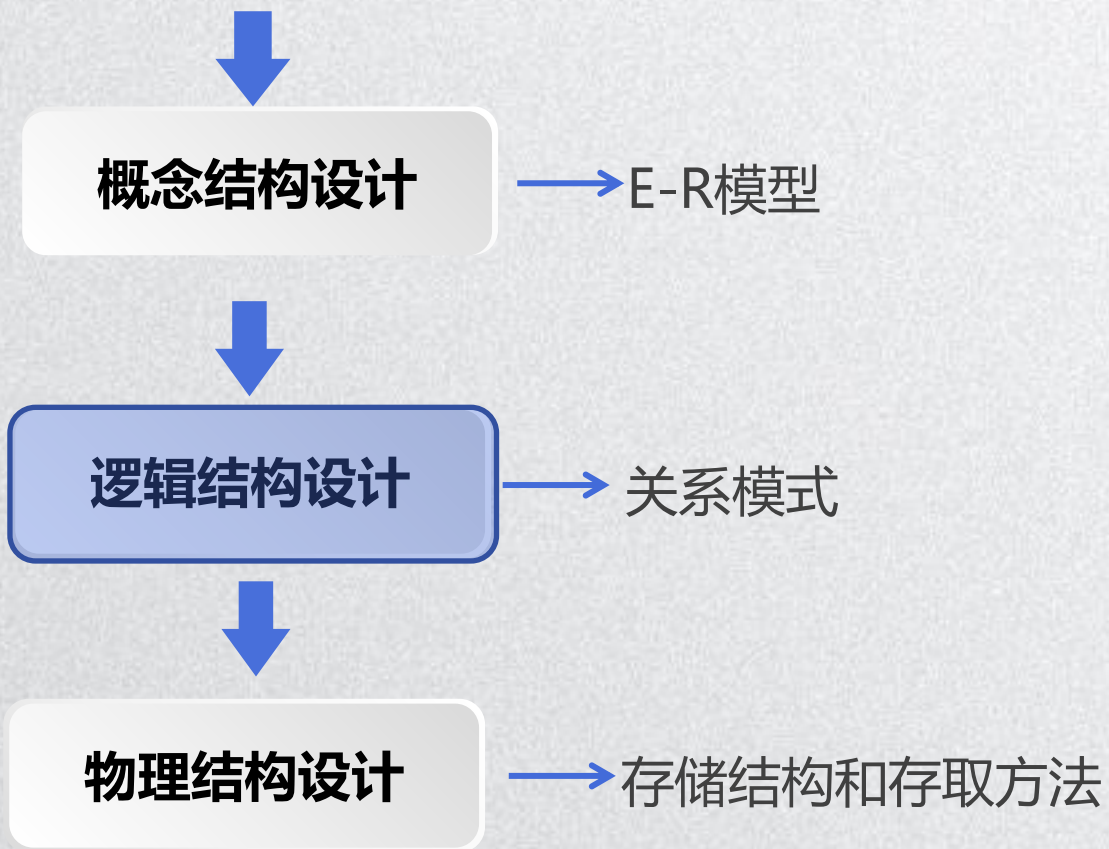
教师（工号，姓名，学院，性别，入校时间，职称，联系电话，照片，个人介绍，密码，是否教务员）

选课（学号，课程号，成绩，录入时间）





# 数据库设计



**THANK YOU!**

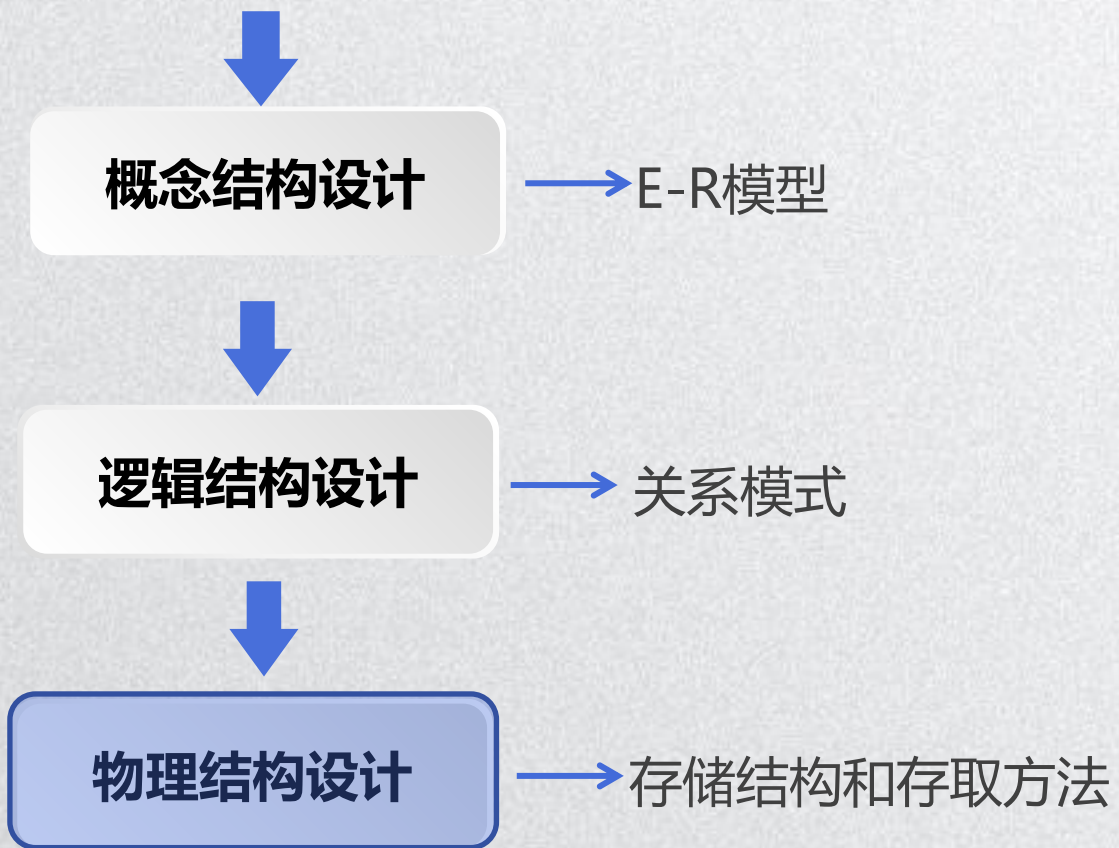


## 9.4 系统设计

### ——数据库物理结构设计



# 数据库设计

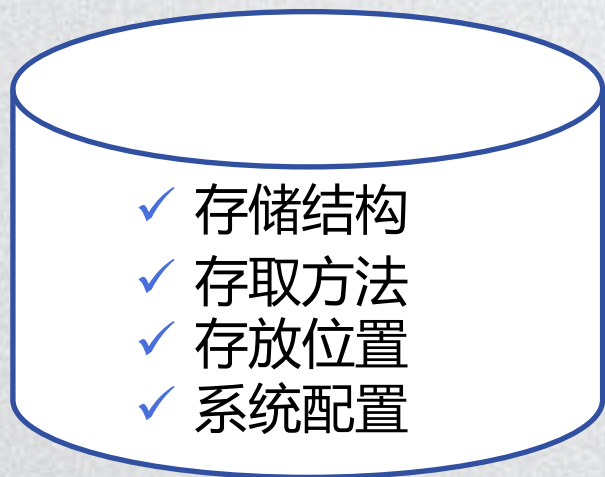






## 数据库物理结构设计

根据数据库逻辑结构，利用选定的DBMS，设计实现数据库。



# 数据库物理结构设计——存储结构

✓ 设计存储结构：特定DBMS下的基本表和视图

学生（学号，姓名，性别，出生日期，是否住校，照片，电话，密码，介绍）

字段名称	字段说明	类型定义	属性约束	数据类型定义说明
StudentCode	学号	char(4)	Primary Key	学号虽然由数字构成，但它由特定含义的一些位构成，最好使用字符型，而且由于一个学校的学号长度是固定的(例如4位)，如果不含有中文字符，使用ANSI字符集合，最合理的设计是定义为char(4)
StudentName	学生姓名	varchar(20)	Not Null	由于长度不固定、且含有中文字符，如果有少数民族学生名字较长(例如最长20字)，可以考虑设计为varchar(20)
Sex	性别	char (1)	Not Null "男"or"女"	取值为中文“男”或“女”，设计为char(1)
Birthday	出生日期	date		设计为date即可
LiveInDorm	是否住校	char(1)	Default 1	取值为0或1，设计为char(1)，默认为1，表示住校
Telephone	联系电话	varchar (40)		存放用“,”分隔的多个电话，长度不固定，为varchar (40)
Photo	照片	varchar(50)		存放照片图象文件所在的路径，设计为varchar(50)
Description	个人介绍	text		是一段文本，设计为Text。
PassWord	密码	varchar(16)		字符串，为安全起见，长度稍长些且不固定，设计为varchar(16)



## 数据库物理结构设计——存取路径

---

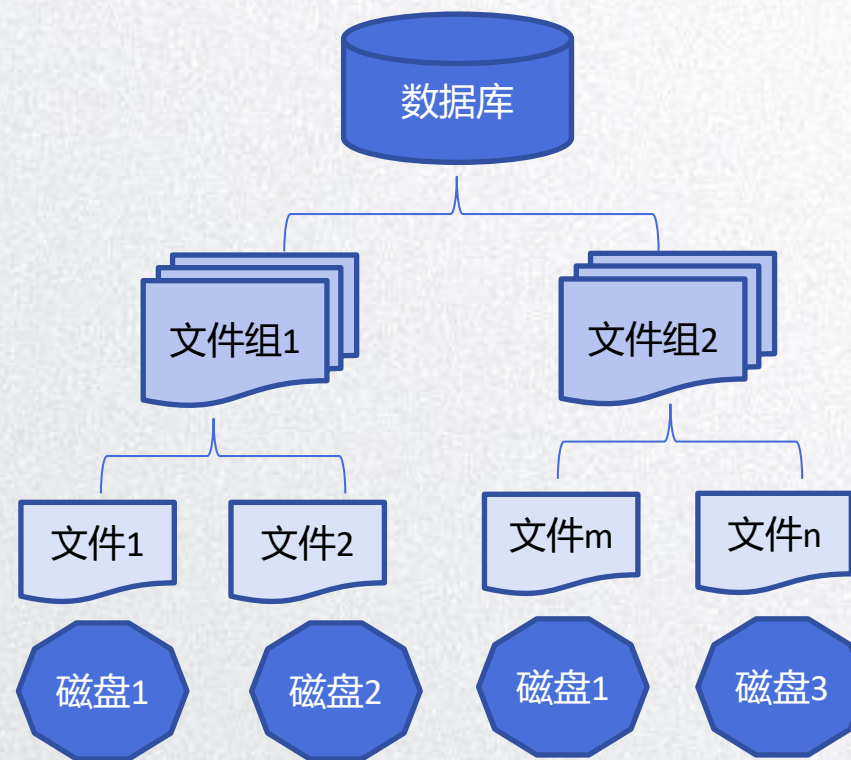
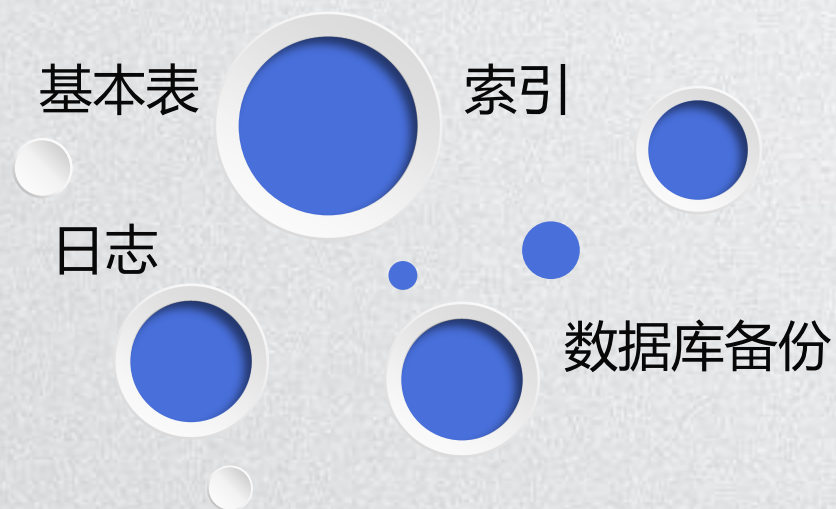
- ✓ 设计存取路径：根据快速查询需求在一些属性上建立索引
  - 主键和外键
  - 检索频繁的字段
  - 经常需要排序的字段

例如：在学生表的“姓名”上建立索引可支持按姓名的快速查询。

学生（学号，姓名，性别，出生日期，是否住校，照片，电话，密码，介绍）

## 数据库物理结构设计——存放位置

- ✓ 设计存放位置：根据各类数据应用特点选择合适的物理存储位置





## 数据库物理结构设计——系统配置

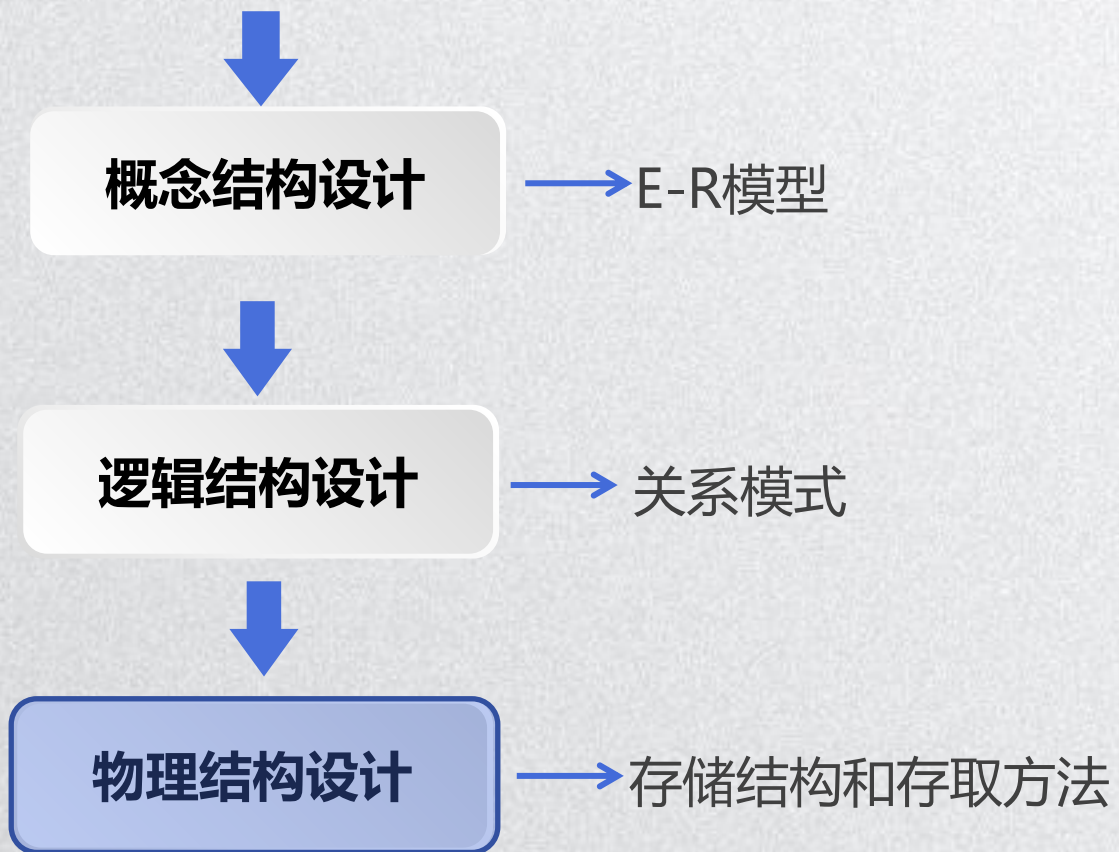
---

- ✓ 确定系统配置：设置DBMS系统配置参数优化运行环境，改善系统性能





# 数据库设计





**THANK YOU!**

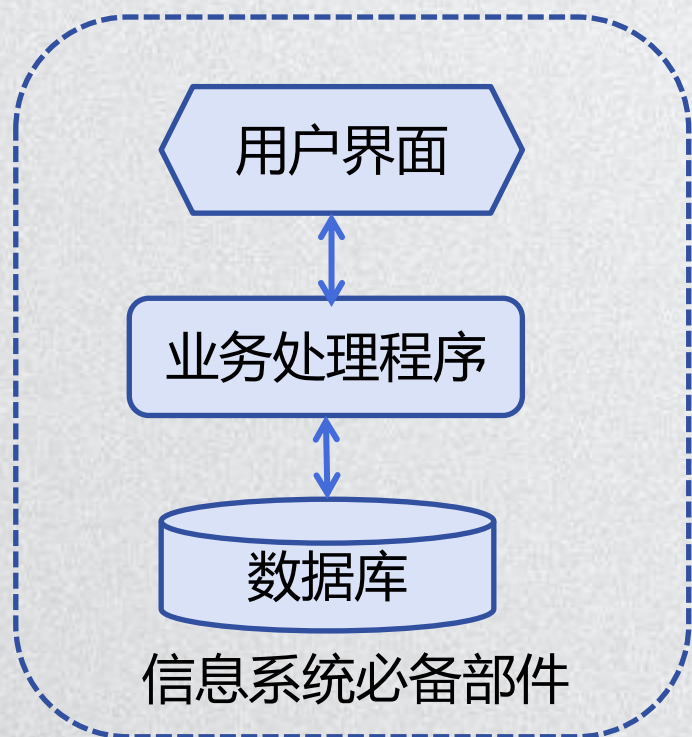
## 9.4 系统设计

### ——系统架构设计





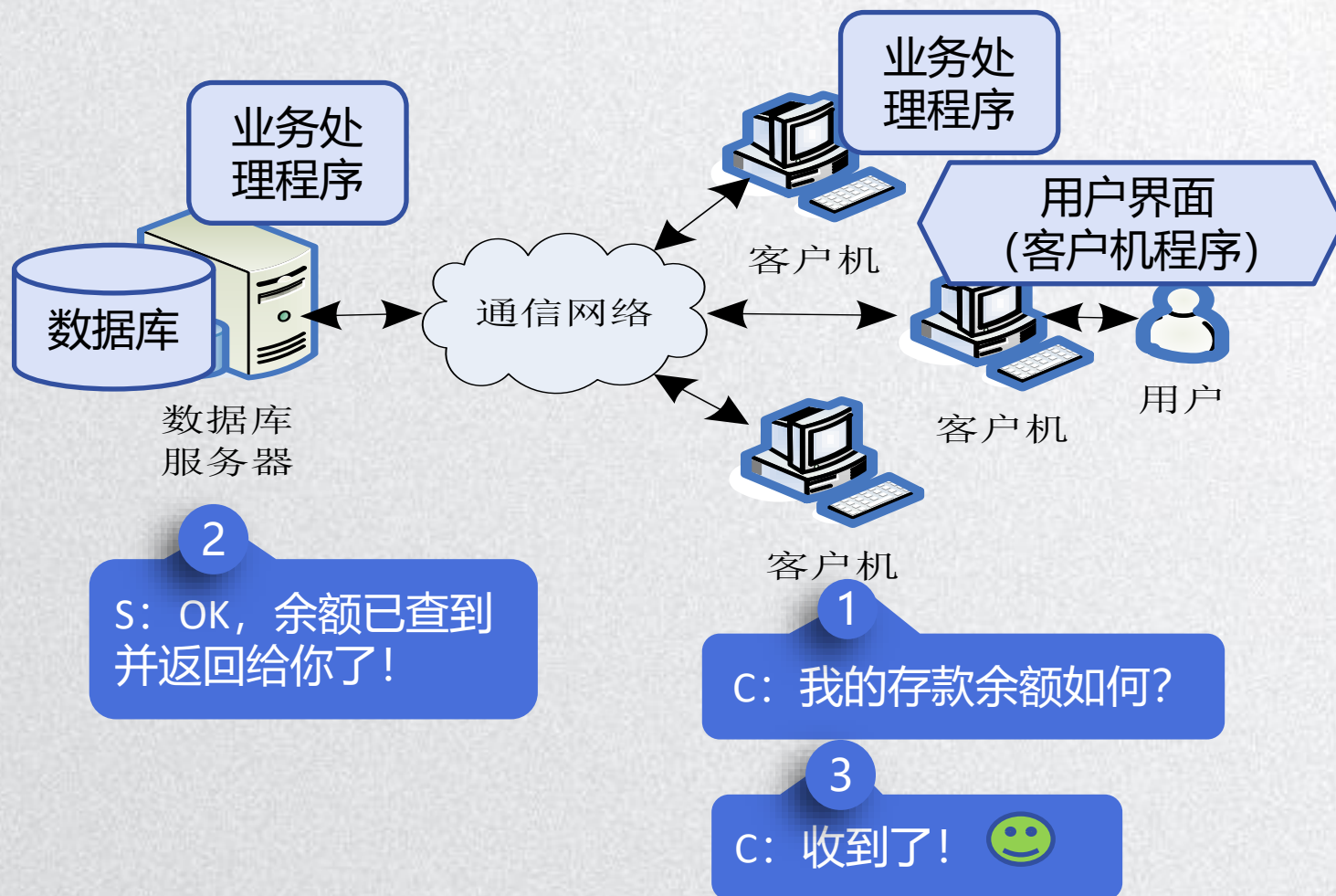
## 系统架构设计



**系统架构**描述构成信息系统的各部分在系统内的分布和协作方式。

# 客户机/服务器模式

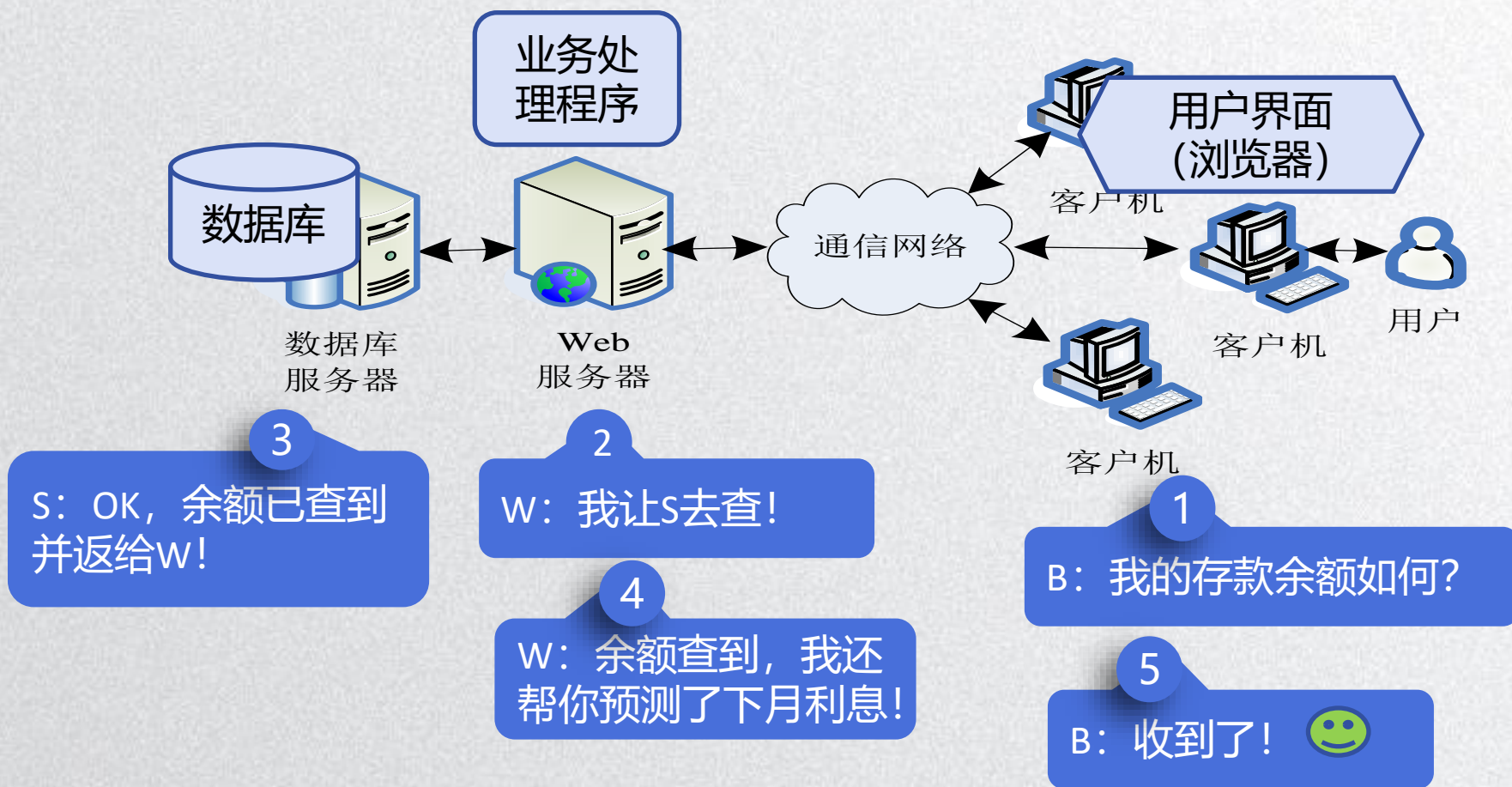
客户机/服务器模式（Client/Server, C/S模式）是面向局域网的两层结构。





# 浏览器/服务器模式

浏览器/服务器（Browser/Server, B/S模式）是一种面向Internet的三层结构。



## B/S和C/S模式的比较

都是基于网络的分布式计算模式。应用程序与数据隔离，计算任务多角色分担。

### C/S结构

适用于局域网应用、客户机数量不多、客户端处理复杂、对系统响应速度和安全性要求高的系统。

- ✓ 实时处理能力强
- ✓ 客户机要装专门软件，维护和升级困难，但增强了系统安全性
- ✓ 系统开放性差，难集成其他服务

VS

### B/S结构

适用于广域网或局域网、客户数量大且分散的系统。

- ✓ 客户端只需Web浏览器，维护扩展方便，而且界面统一
- ✓ 系统开放性强，易集成其他服务，但增加了受攻击的风险



## 其他新型模式

- **PtoP模式 (Peer to Peer, 简称PtoP或P2P)**

即**对等网络结构**。取消了服务器的中心地位，系统内计算机通过数据交换直接共享资源和服务。

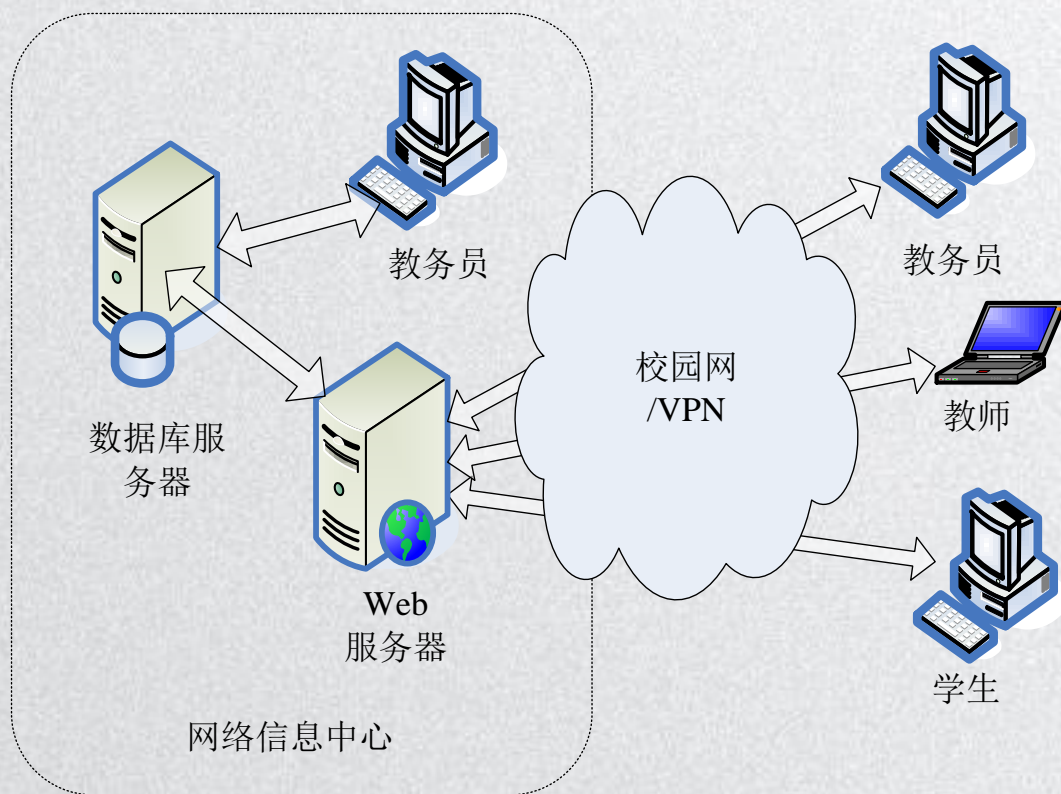


- **云计算**

基于Web的分布式计算，可将Internet上分布的大量计算机连接成虚拟资源，协同工作，为用户按需提供各种IT服务。



## 举例：教务系统架构设计及开发环境



混合架构  
(总体B/S, 个别模块C/S)

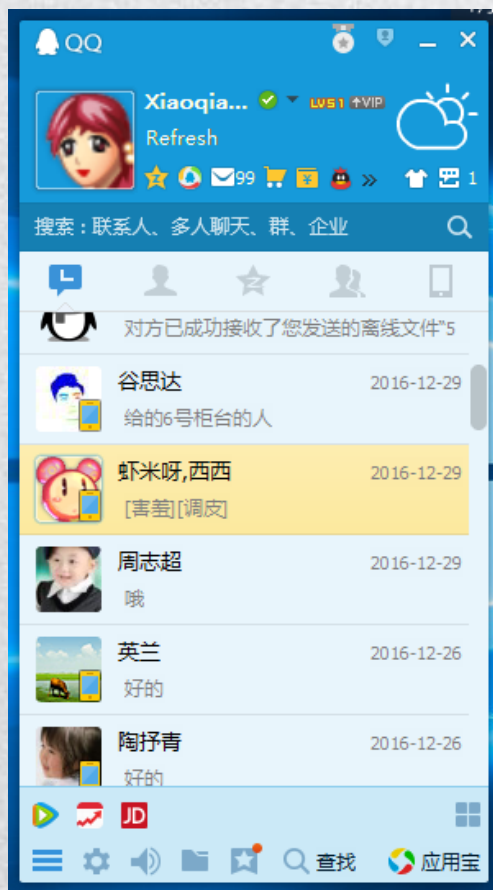
数据库: SQL Server  
Web服务器: Microsoft IIS

前台开发工具:  
B/S程序: ASP.NET和C#语言  
C/S程序: VB.NET或C#语言

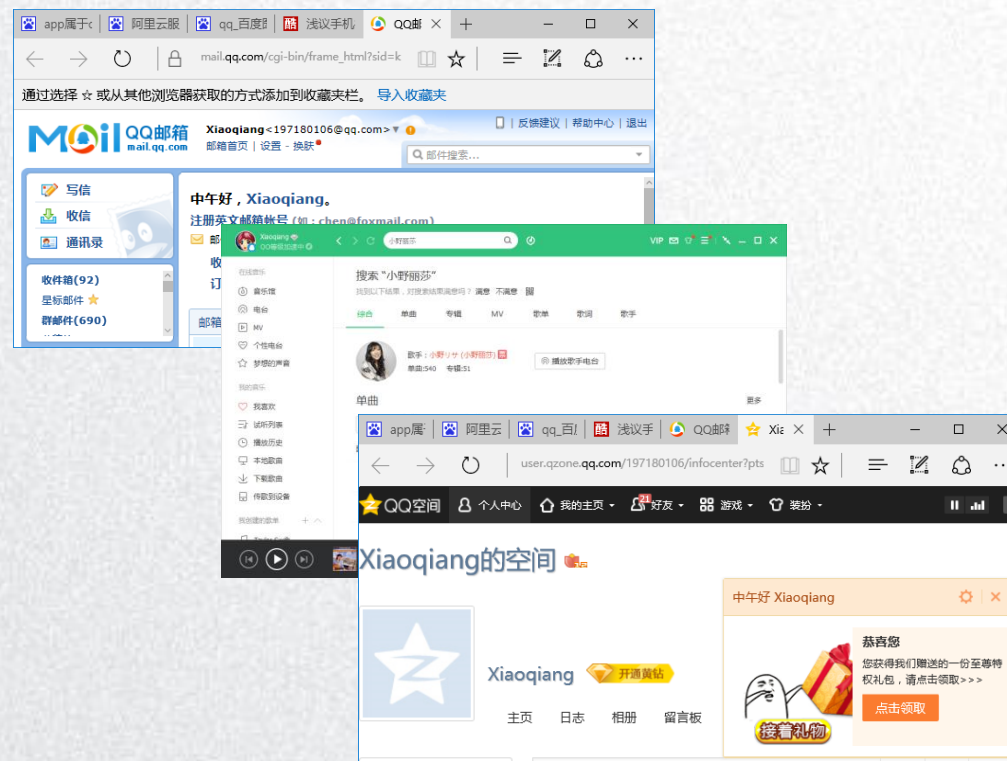
网络和容灾设计:  
VPN虚拟专网  
服务器磁盘RAID  
系统备份方案



# 举例：QQ应用



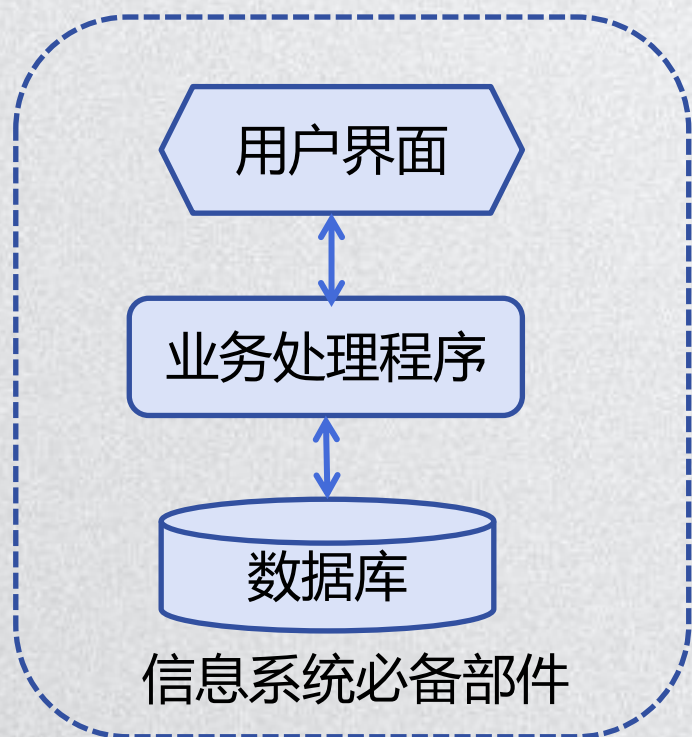
IMS采用C/S+P2P



其他应用采用B/S



## 系统架构设计



**系统架构**描述构成信息系统的各部分在系统内的分布和协作方式。



**THANK YOU!**

## 9.4 系统设计

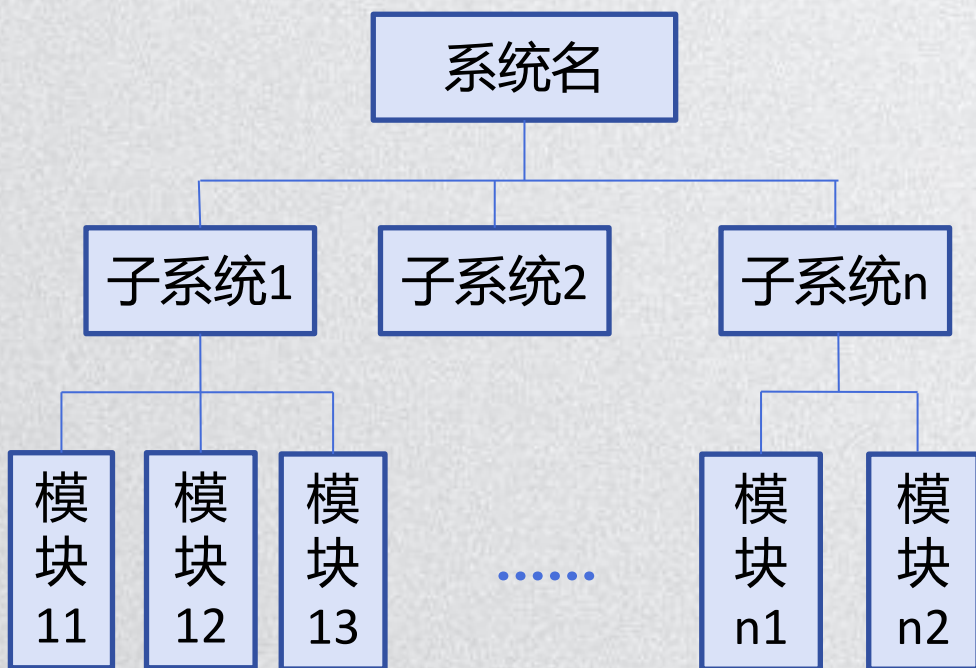
### ——系统功能结构设计



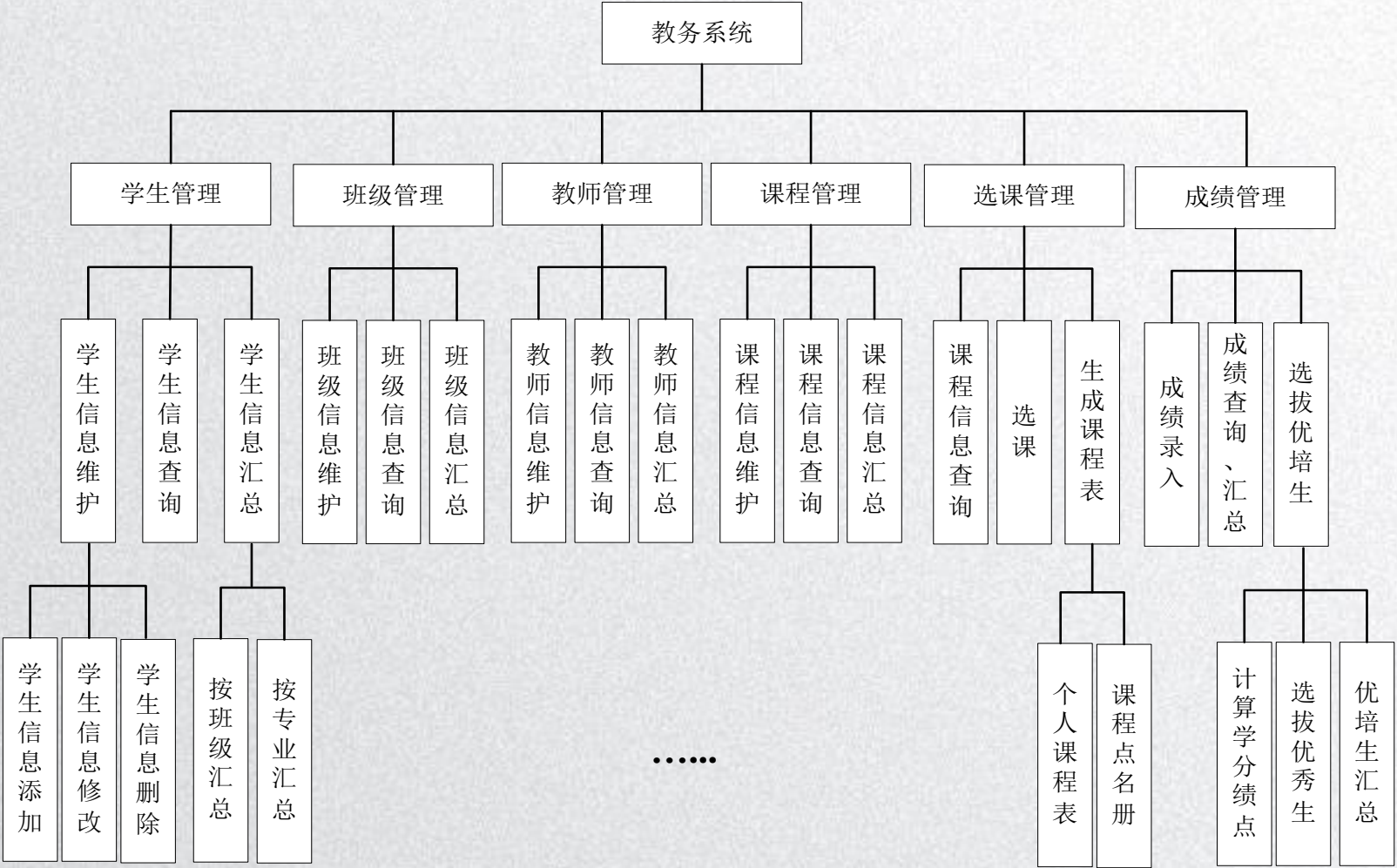


## 系统功能结构设计

系统功能结构图：将系统的功能进行分解，按功能从属关系表示。

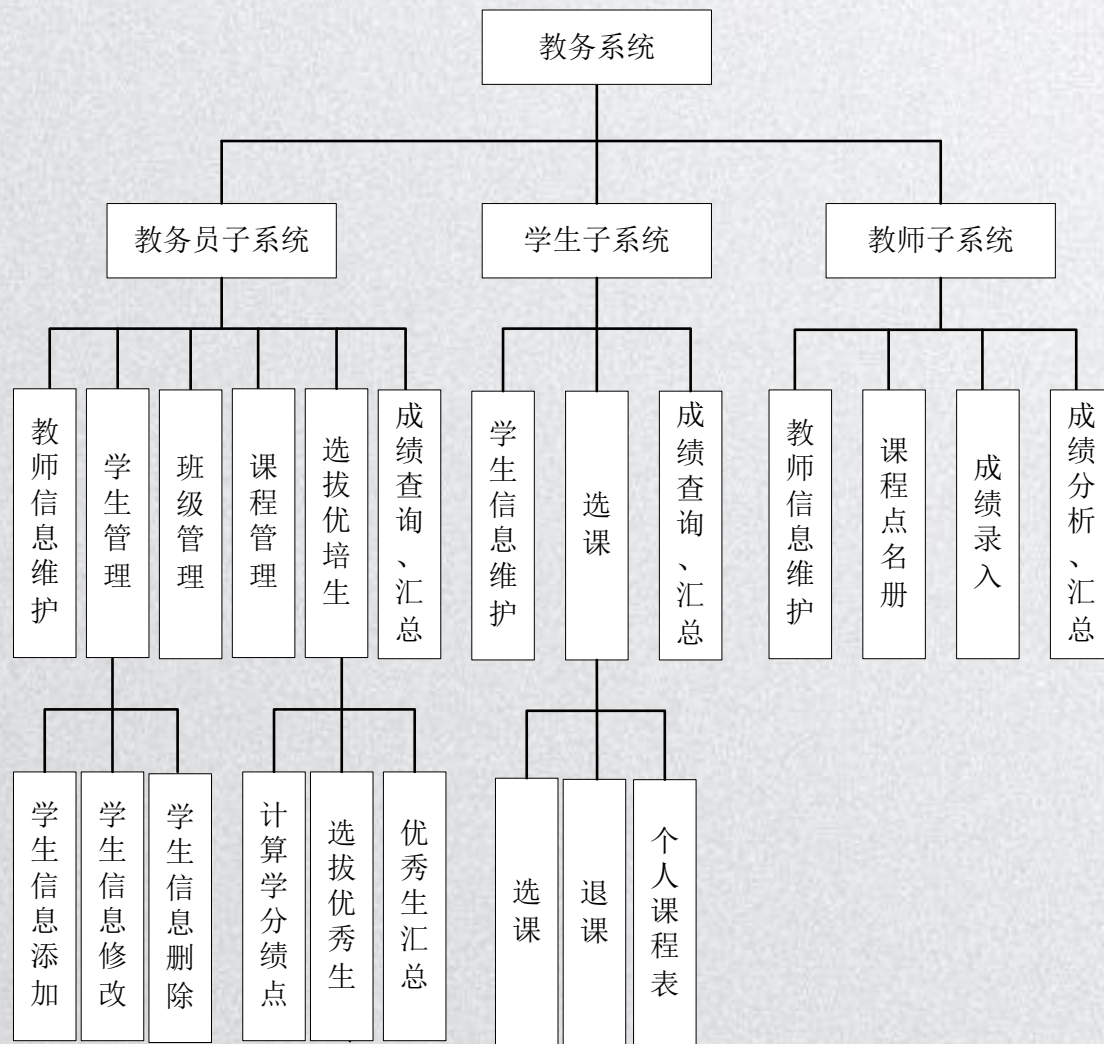


# 举例：教务系统功能结构图

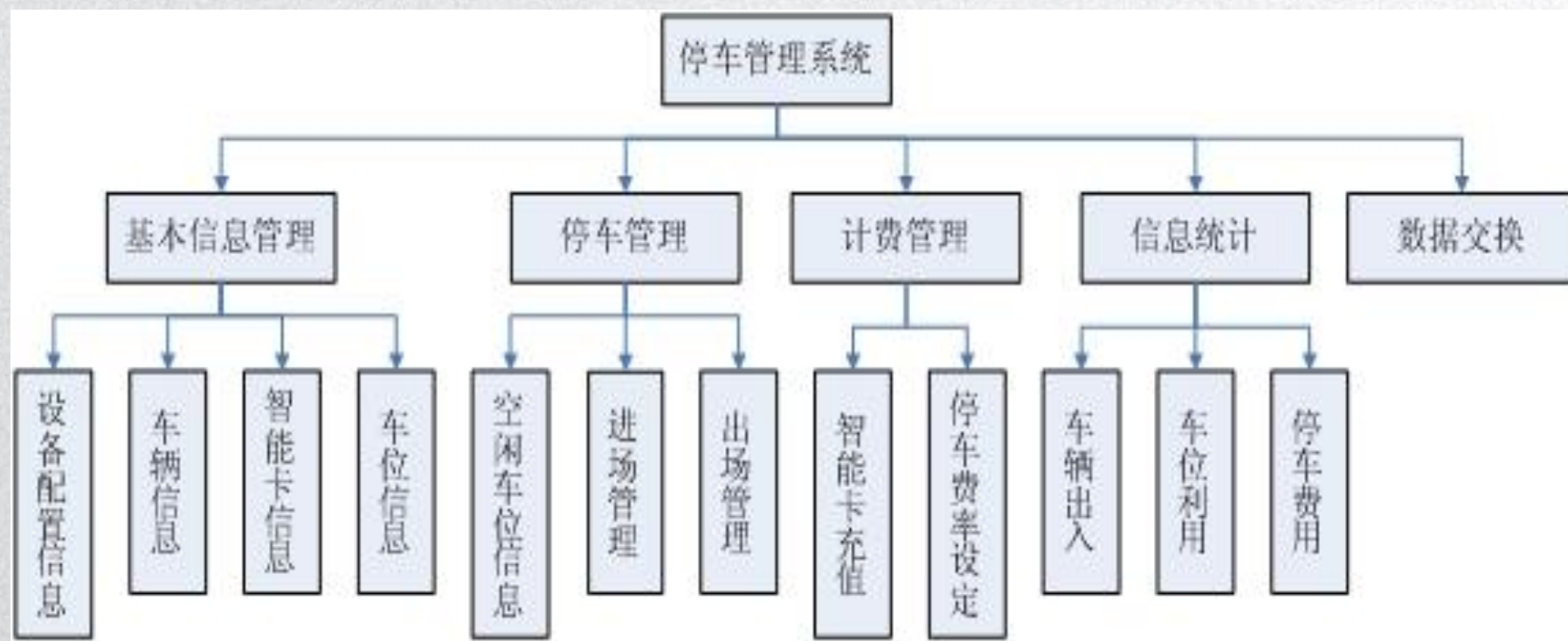




## 举例：教务系统功能结构图



## 举例：停车系统功能结构图

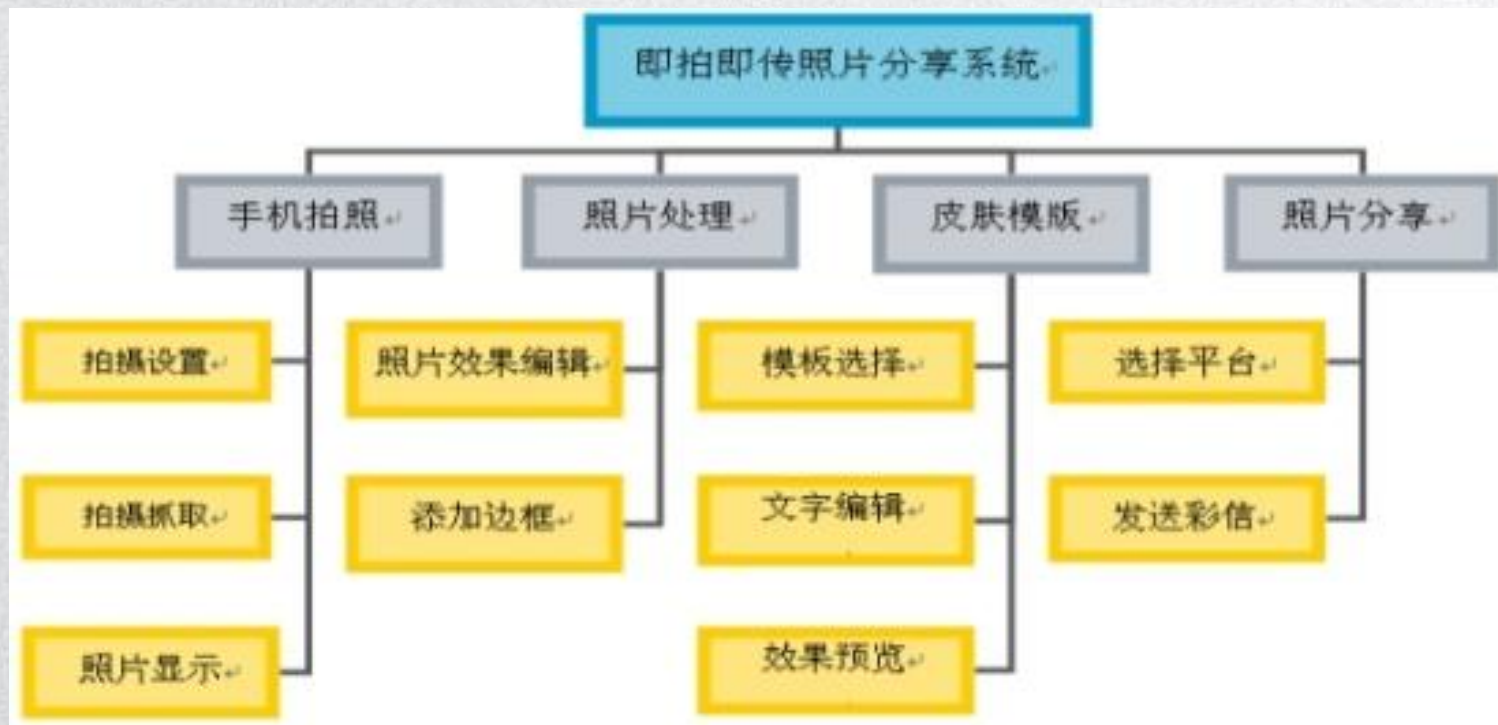




# 举例：微信系统功能结构图



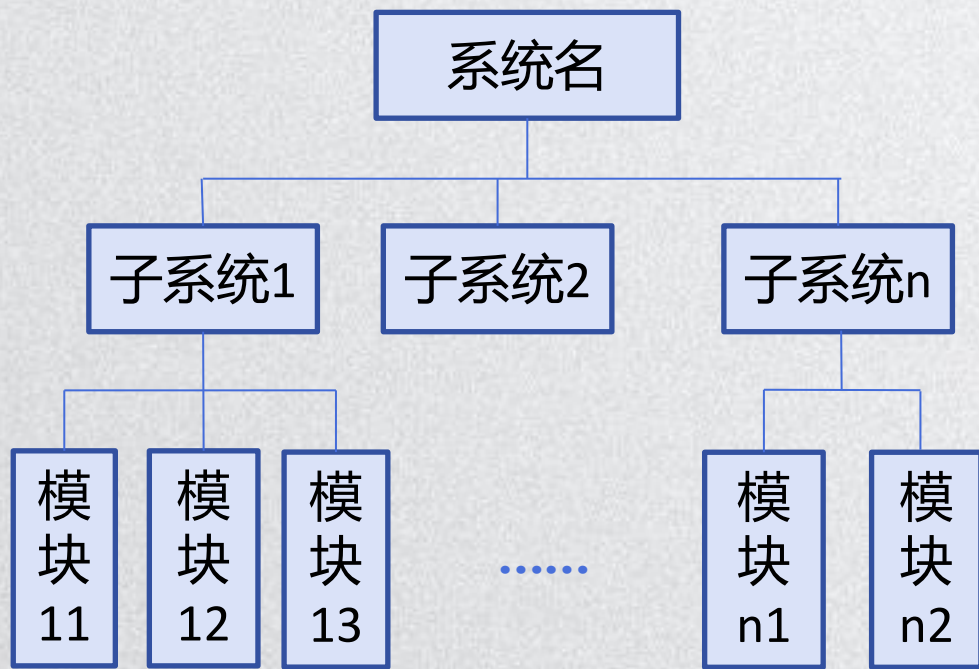
## 举例：手机照片分享系统功能结构图







# 系统功能结构设计



系统功能结构示意图

**THANK YOU!**



## 9.4 系统设计

### ——用户界面设计

## 用户界面设计

是用户使用信息系统的综合操作环境，  
由各种菜单、功能、输入、输出以及影响  
人机通信的一些硬件和软件特性组成。



用户界面 (User Interface, UI)



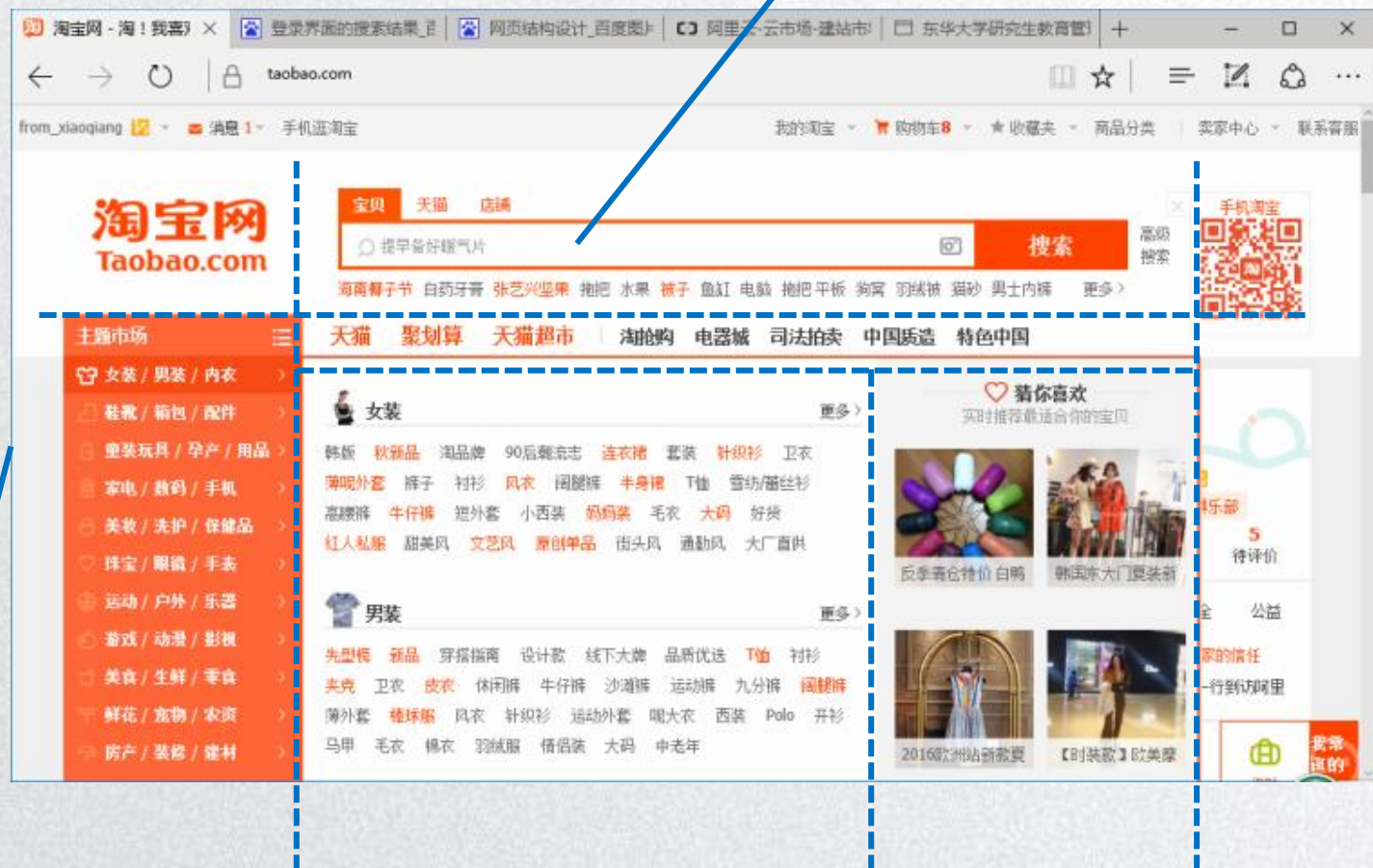
# 用户界面设计的内容

## 联想式关键词查询

✓ 结构设计

✓ 交互设计

✓ 视觉设计



整体明艳的橘红色

淘宝网主题市场页面

# 用户界面设计的原则

- ✓ 友好明确的人机交互
- ✓ 减少输入，保证数据质量
- ✓ 精确的显示和输出

企汇网  
QIHUIWANG.COM

下午好：郭晓 (产品部 > 设计专员) 账户管理 退出  
7月16日 星期二 农历六月初九

个人桌面 工作流 个人事务 人力资源 行政办公

功能菜单

- 档案查询
- 合同管理
- 奖惩管理
- 人事调动
- 考勤统计
- 工作日管理
- 社保

手机号码	手机类型	部门	姓名	性别	领取时间	申请原因	备注	操作
15639919875	中兴V880	产品部	石头	女	2013-05-26	工作需要	备注	修改
15639919875	王五 2013-02-02—2013-05-22	客服部	女	2013-05-26	工作需要	备注	修改	
15639919875	招五 2013-02-02—2013-05-22	销售部	女	2013-05-26	工作需要	备注	修改	
15639919875	王五 2013-02-02—2013-05-22	客服部	女	2013-05-26	工作需要	备注	修改	
15639919875	中兴V880	产品部	石头	女	2013-05-26	工作需要	备注	修改

手机号码:  查询 添加

手机添加

手机号:

部门:

姓名:

性别: ☒ 男 ☐ 女

领取时间:

申请原因:

备注:

提交 取消

Copyright ©2012 河南企汇网信息技术有限公司

原则1：以用户为中心



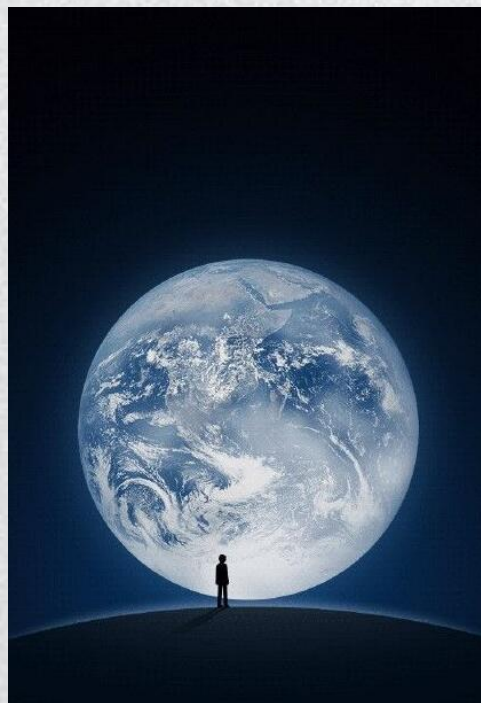
# 用户界面设计的原则

- ✓ 友好明确的人机交互
- ✓ 减少输入，保证数据质量
- ✓ 精确的显示和输出

12306火车订票

原则1：以用户为中心

# 用户界面设计的原则



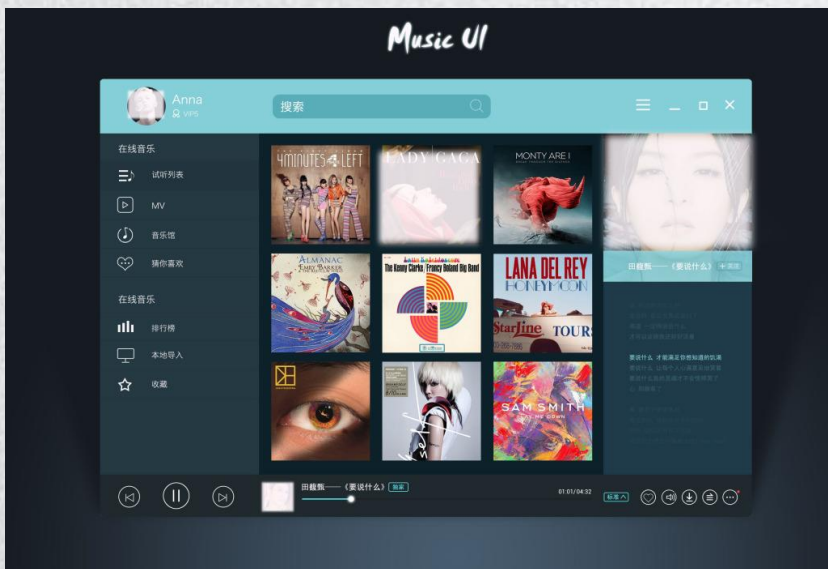
原则2: KISS -- "Keep It Simple And Stupid"





# 用户界面设计

是用户使用信息系统的综合操作环境，由各种菜单、功能、输入、输出以及影响人机通信的一些特性组成。



用户界面 (User Interface, UI)

## 9.4 系统设计

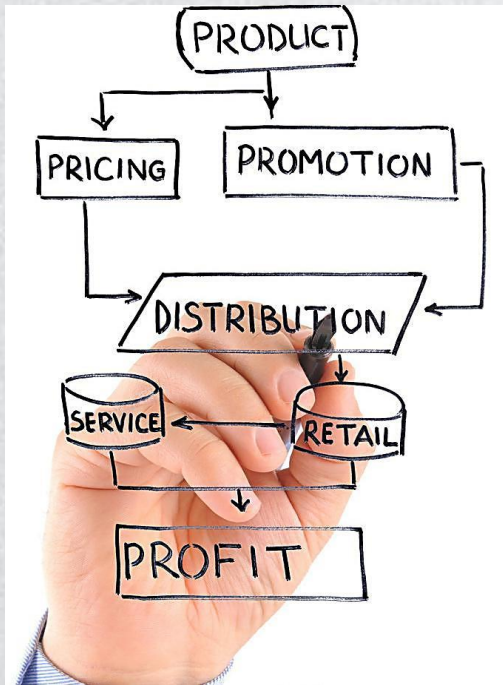
### ——处理流程设计





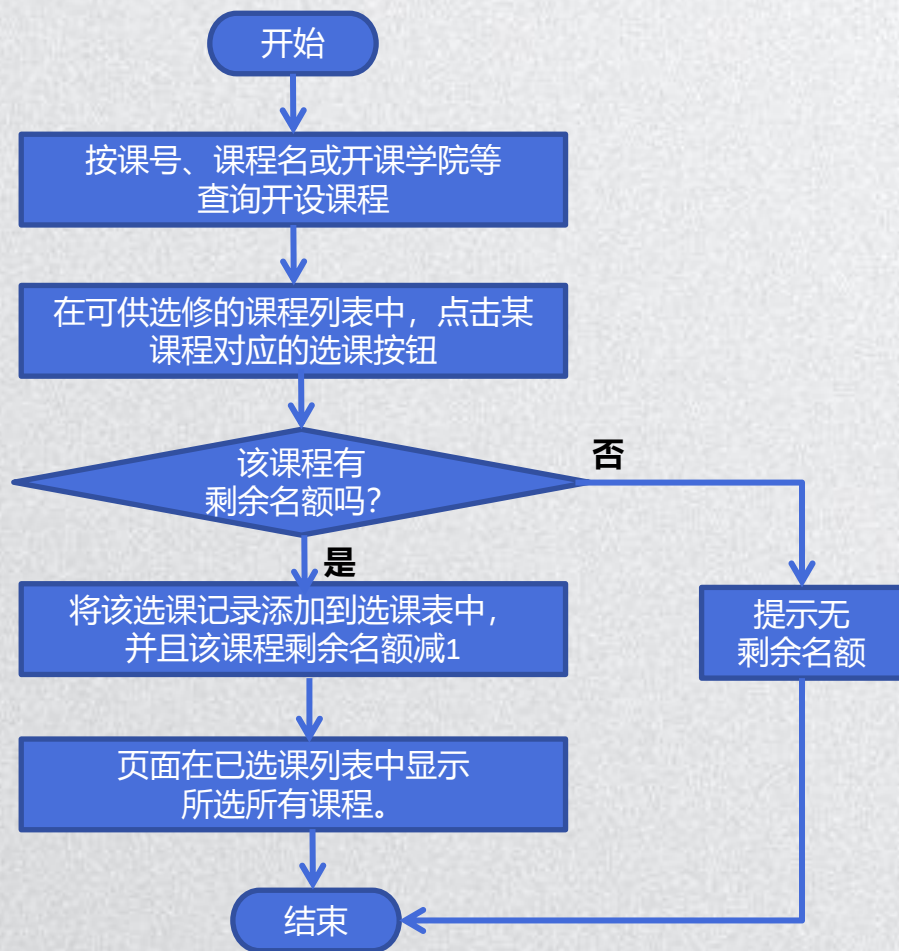
## 处理流程设计

按照系统功能要求，确定计算机处理流程和相关数据存储需求。



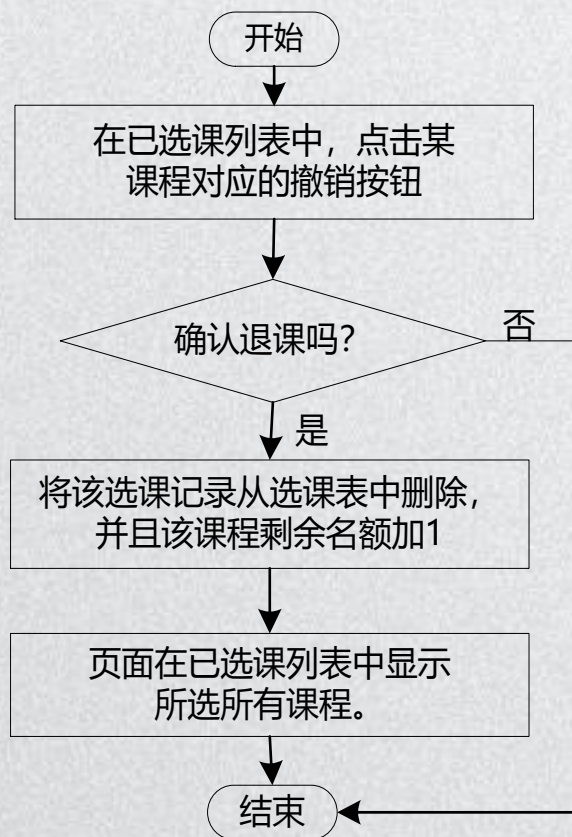
流程图

## 举例：教务系统选课处理流程图

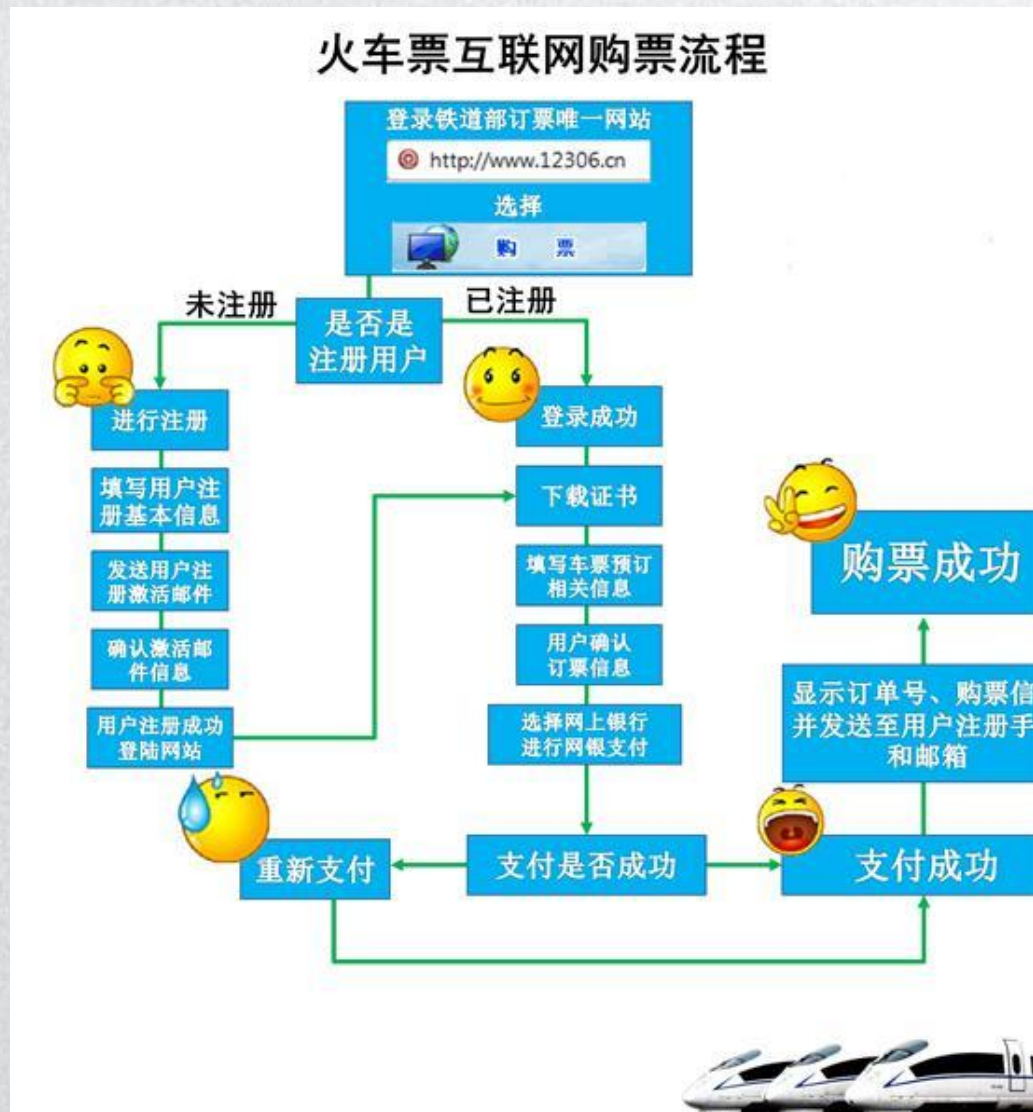




## 举例：退课处理流程图



# 举例：网购火车票流程图

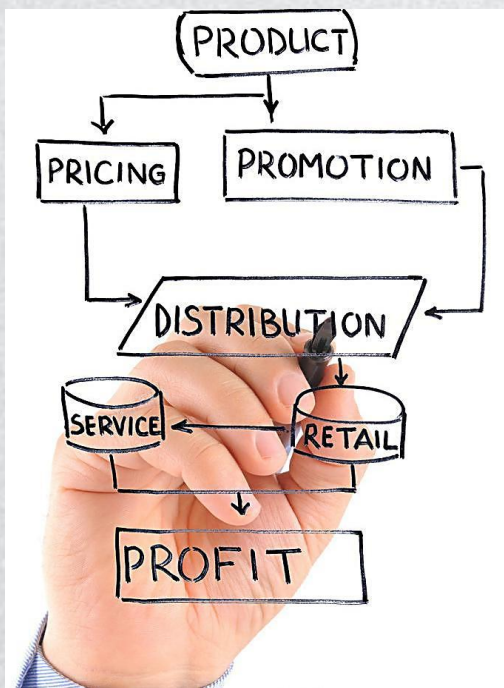






## 处理流程设计

按照系统功能要求，确定计算机处理流程和相关数据存储需求。



流程图

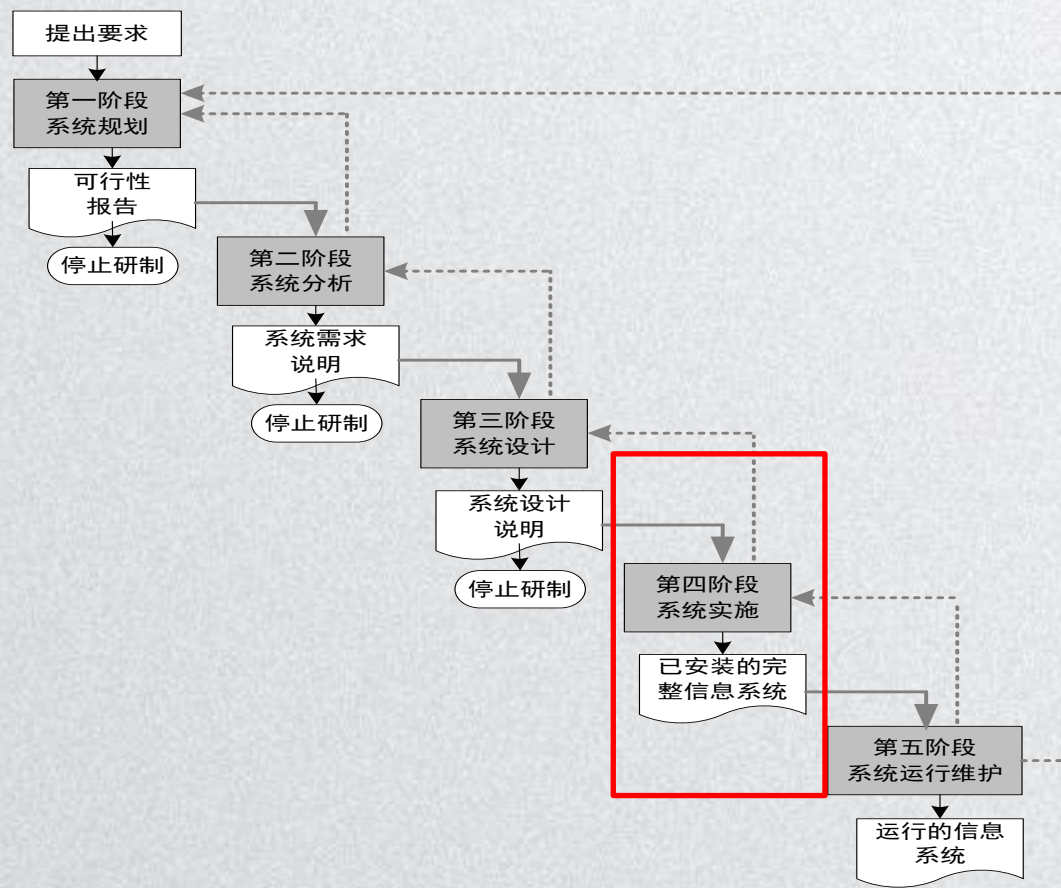
**THANK YOU!**



## 9.5 系统实施



# 系统实施







# 系统实施



# 系统实施

## 程序开发

根据系统设计说明书，建立实际数据库结构，用程序设计语言，编制与调试应用程序，实现系统功能。





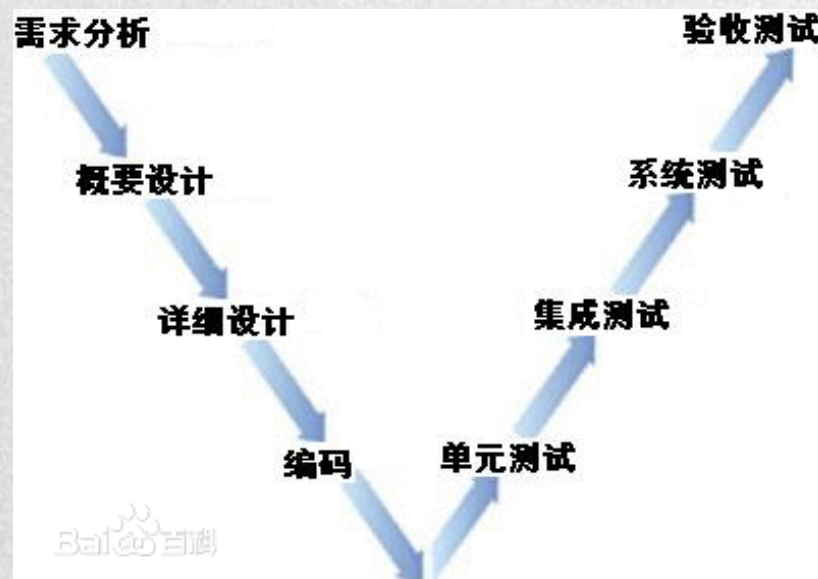
# 系统实施



验证系统是否满足需求规格定义。不仅要测试软件，还要测试数据库、网络、硬件、外设、某些支持软件及其接口等。

功能测试

性能测试



# 系统实施

## 文档整理

对系统开发过程中产生的所有文档进行整理和完善，形成随系统一同交付的文档。文档是今后系统运行和维护的依据。



### <软件设计文档国家标准>

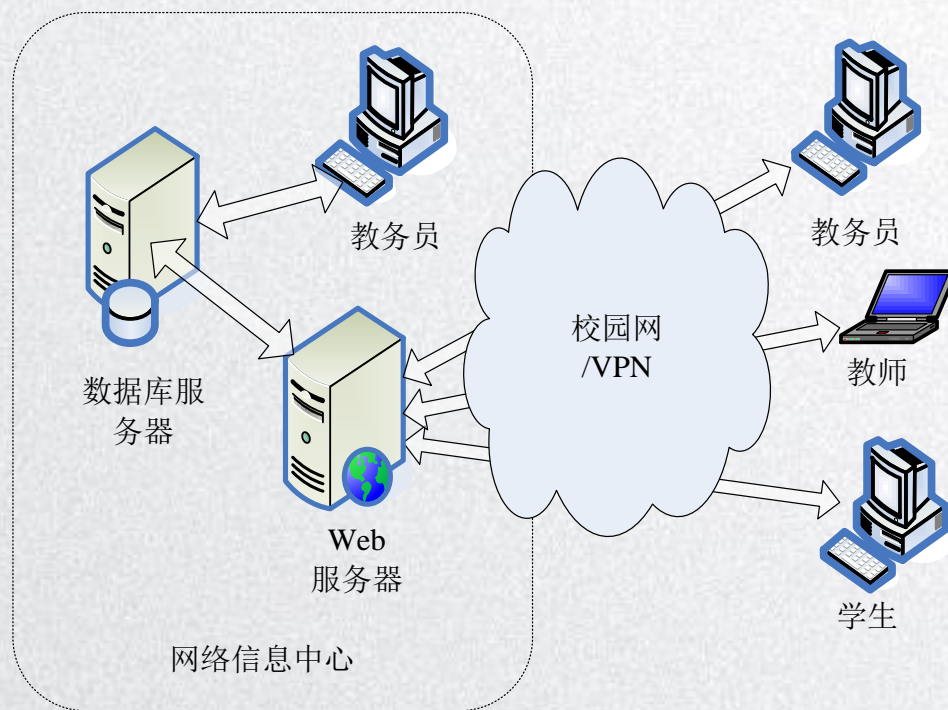
- ├操作手册 (GB8567——88)
- ├测试分析报告 (GB8567——88)
- ├测试计划 (GB8567——88)
- ├概要设计说明书 (GB8567——88)
- ├开发进度月报 (GB8567——88)
- ├可行性研究报告 (GB8567——88)
- ├模块开发卷宗 (GB8567——88)
- ├软件需求说明书 (GB856T——88)
- ├数据库设计说明书 (GB8567——88)
- ├数据要求说明书 (GB856T——88)
- ├文件编制实施规定的实例 (GB8567-88)
- ├详细设计说明书 (GB8567——88)
- ├项目开发计划 (GB856T——88)
- ├项目开发总结报告 (GB8567——88)
- └用户手册 (GB8567——88)



## 系统 安装

安装系统的运行支持环境和信息系统。涉及计算机网络以及一系列专门的硬件和软件的支持。

- 建立系统运行环境
- 安装应用系统
- 数据转换



教务系统部署

# 系统实施

## 系统转换

系统转换是指新系统替换手工工作或原有系统投入在线运行的过程。

- ✓ 直接转换
- ✓ 并行转换
- ✓ 逐步替换

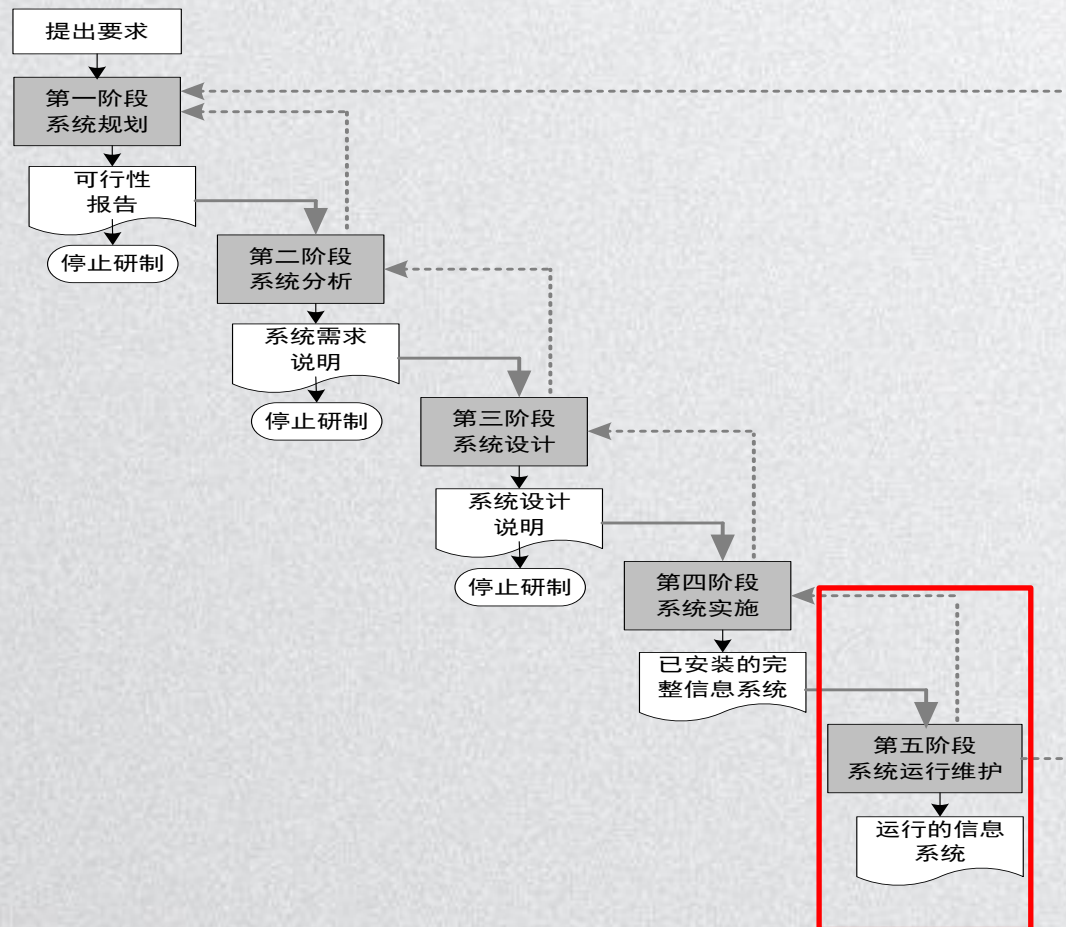




## 9.6 系统运行和维护



# 系统运行和维护







## 系统运行与维护

系统投入运行后，需要持续性对系统进行维护和评价，记录系统运行情况，维护、增强和保护系统。

维护

更改错误或适应功能变化

增强

提供新的特征或优势

保护

通过安全控制免受威胁



**THANK YOU!**