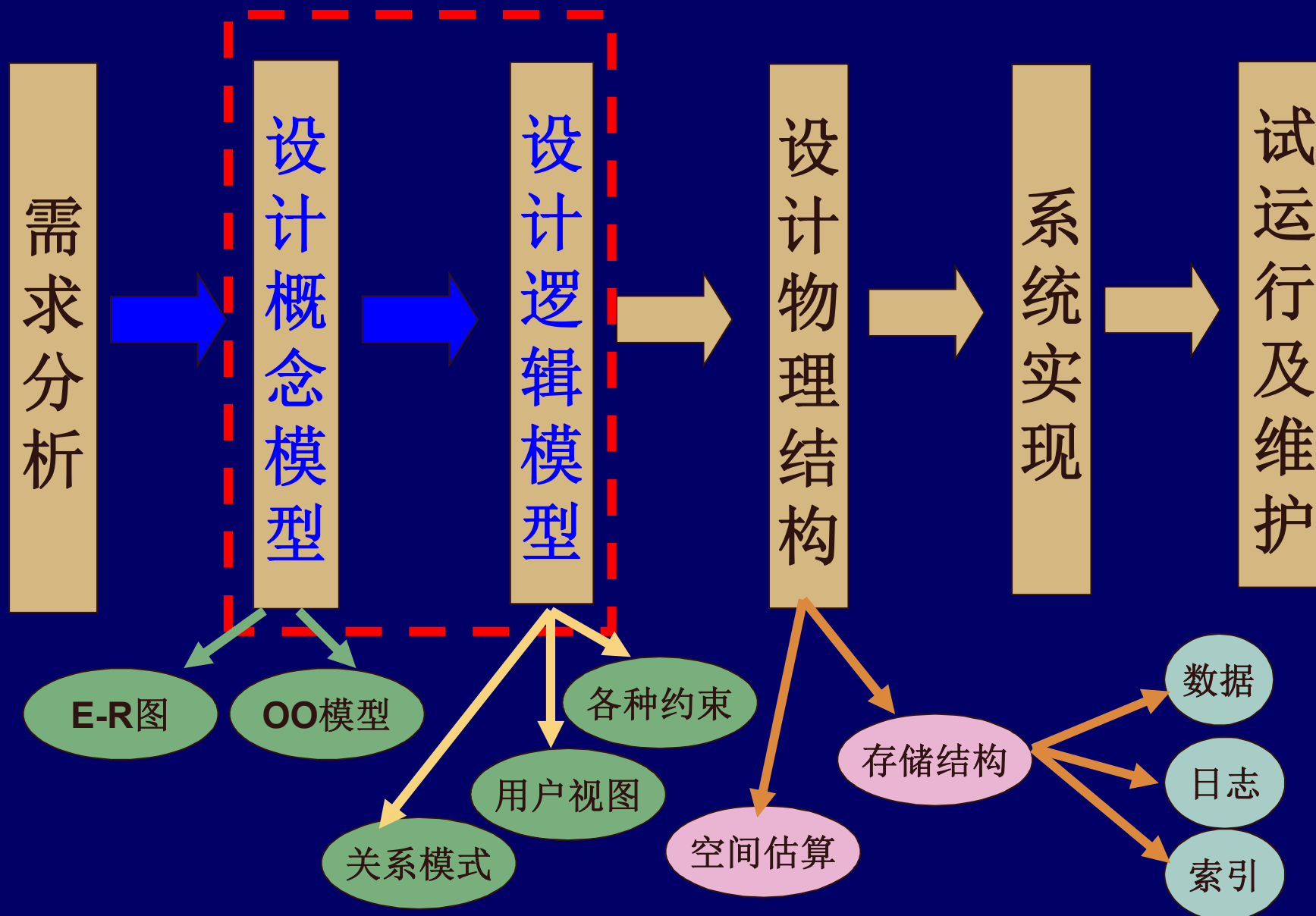


第二讲：数据库设计概述

第二讲：数据库设计概述



1 概念模型的建立

本质：从用户的角度，对系统中数据及数据间关系进行抽象。

表现方法：

- * E-R (EE-R) 模型
- * 00模型
- * 其它模型

抽象机制：

- * 分类—数据
- * 关联—数据间联系
- * 概括/特化—数据间的蕴含关系
- * 聚集/分解—复杂的数据关系
- * 隶属—弱实集

教务管理系统中的数据

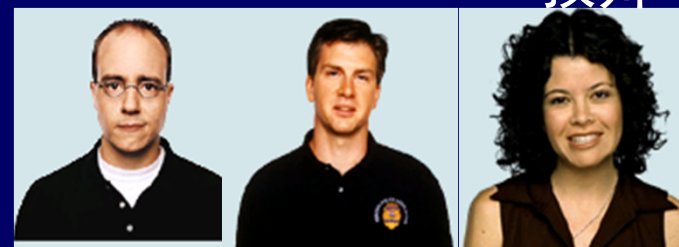


教务管理系统中的数据



学生

学号，姓名，年龄，出生地

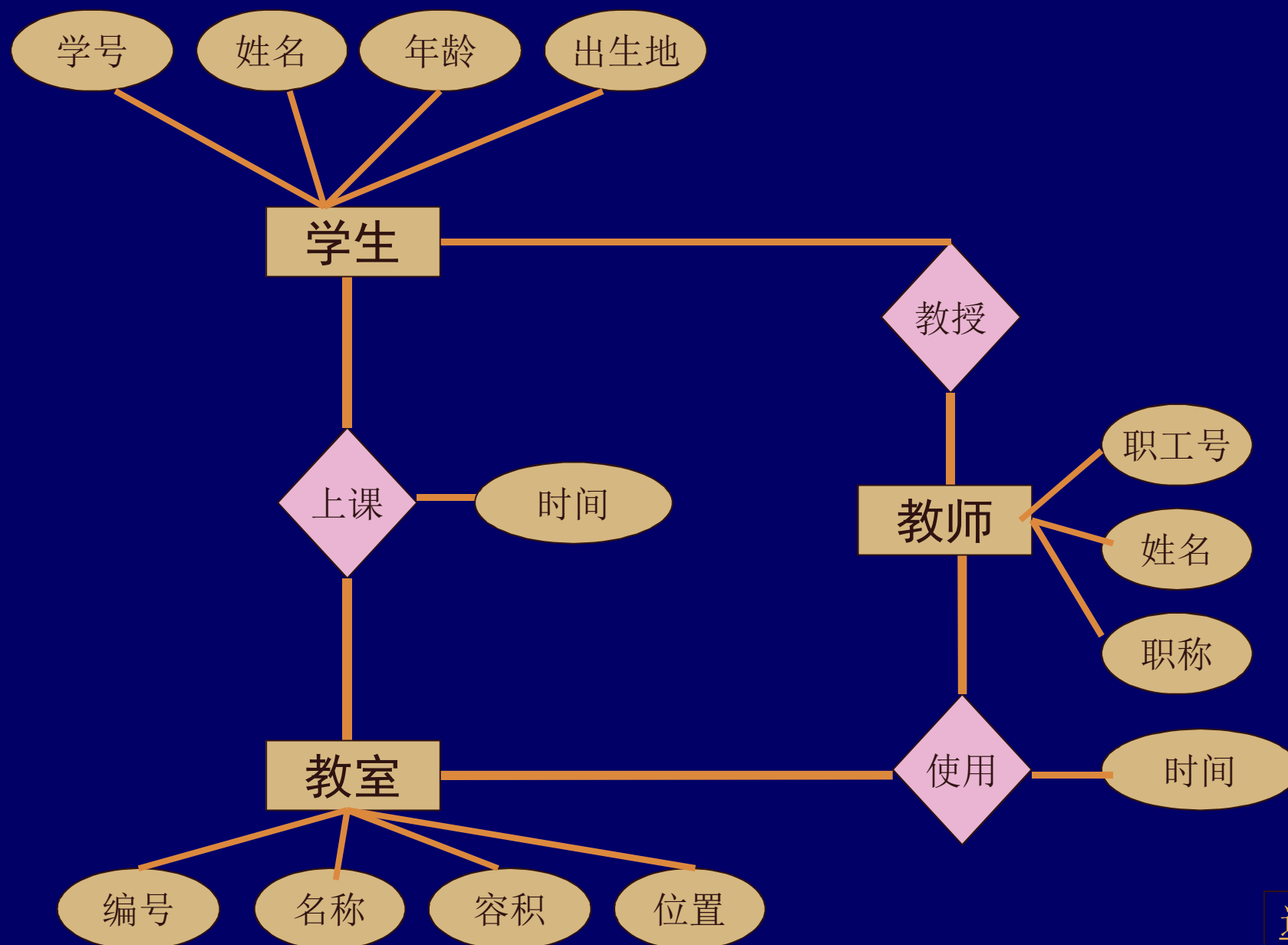


教师

职工号，姓名，年龄，职称



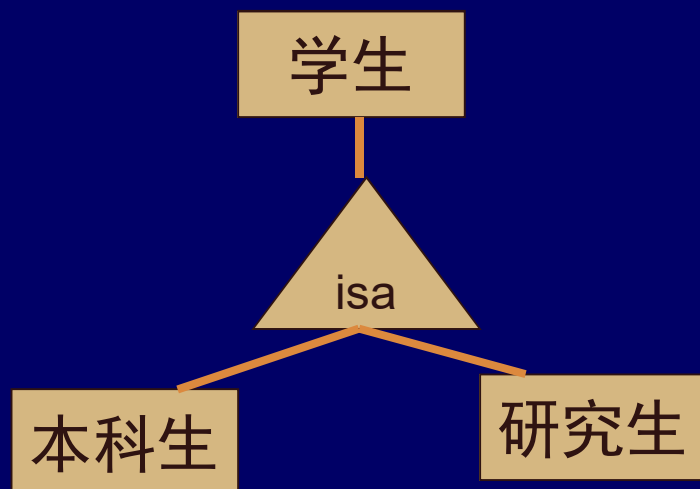
教室



返回

概括：从若干个数据对象类别中抽取共同的特性，形成更高层次的数据对象类别。

特化：按照某些特性把较高层次的数据对象类别区分为若干个数据对象类别子集。

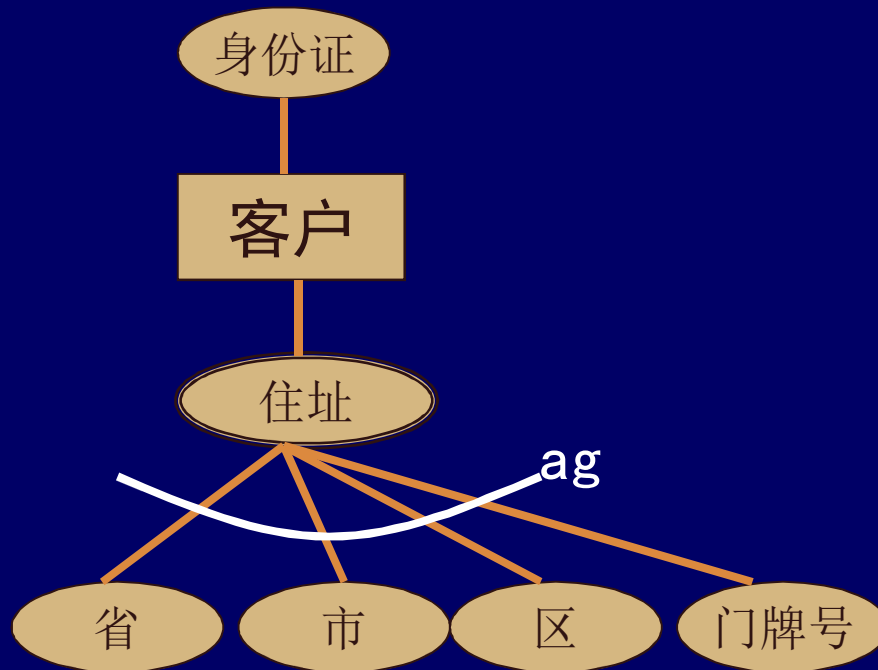


这实际上是子类的表达！

返回

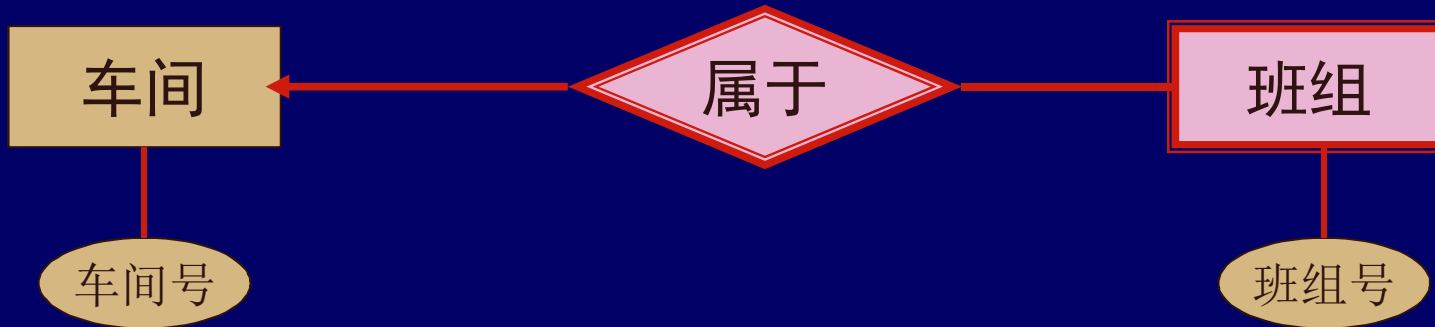
聚集：将若干数据对象类别或特性，关联抽象成为高一层次数据对象类别或特性

分解：将一个数据对象类别或特性，分解成为若干个称为“部分”的数据对象类或特性。



返回

隶属：两类数据间有属于的关系，且一类数据无标识，需要靠另一类数据帮助识别。

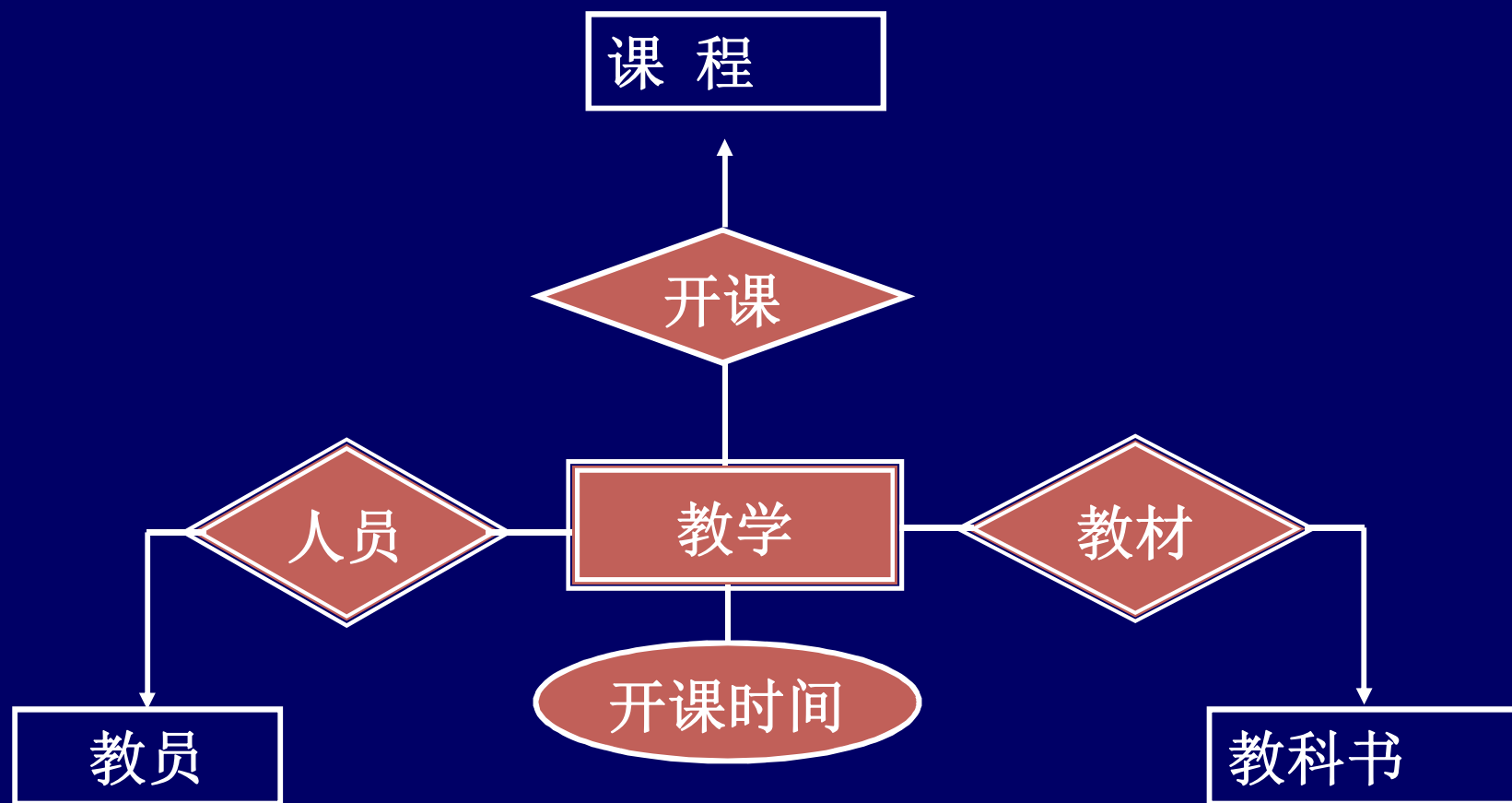


(一车间，二车间，三车间...)

一班，二班，三班
一班，二班，三班

...

象这种属于某一类数据又不能独立标识自己的数据类称为弱实体集。弱实体还可以来自多路联系。



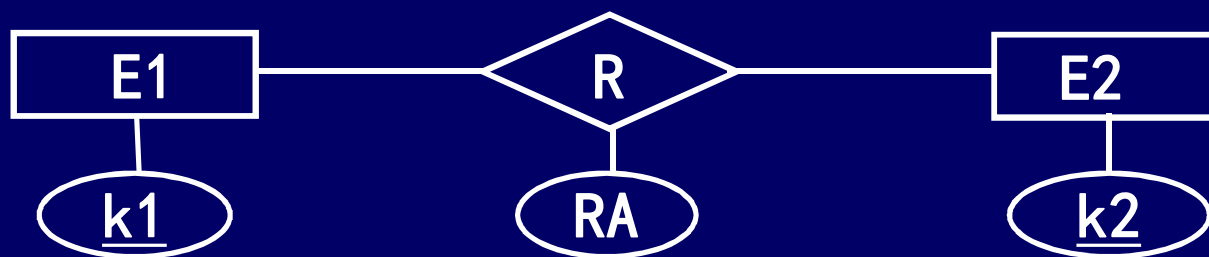
返回

2 逻辑模型的建立

逻辑模型针对具体的数据库管理系统，目前主流关系型DBMS，所以要熟练掌握E-R向关系模型的转化！

方法：

- * 变换实体集
- * 变换联系



$R(k1, k2, RA)$

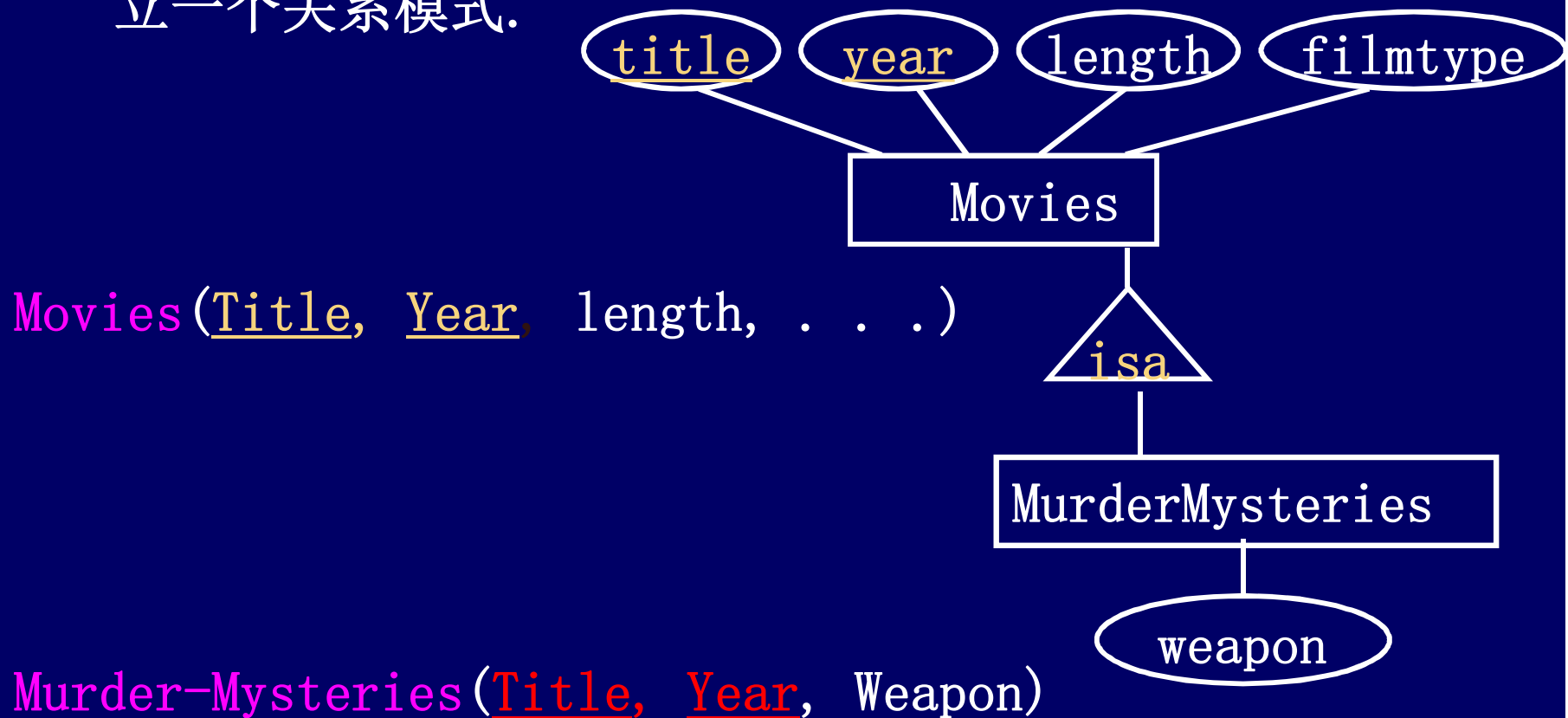
正确确定键码！

* 子类实体集的变换

“ISA” 不用转化，只涉及到高一层实体与低一层实体如何转化

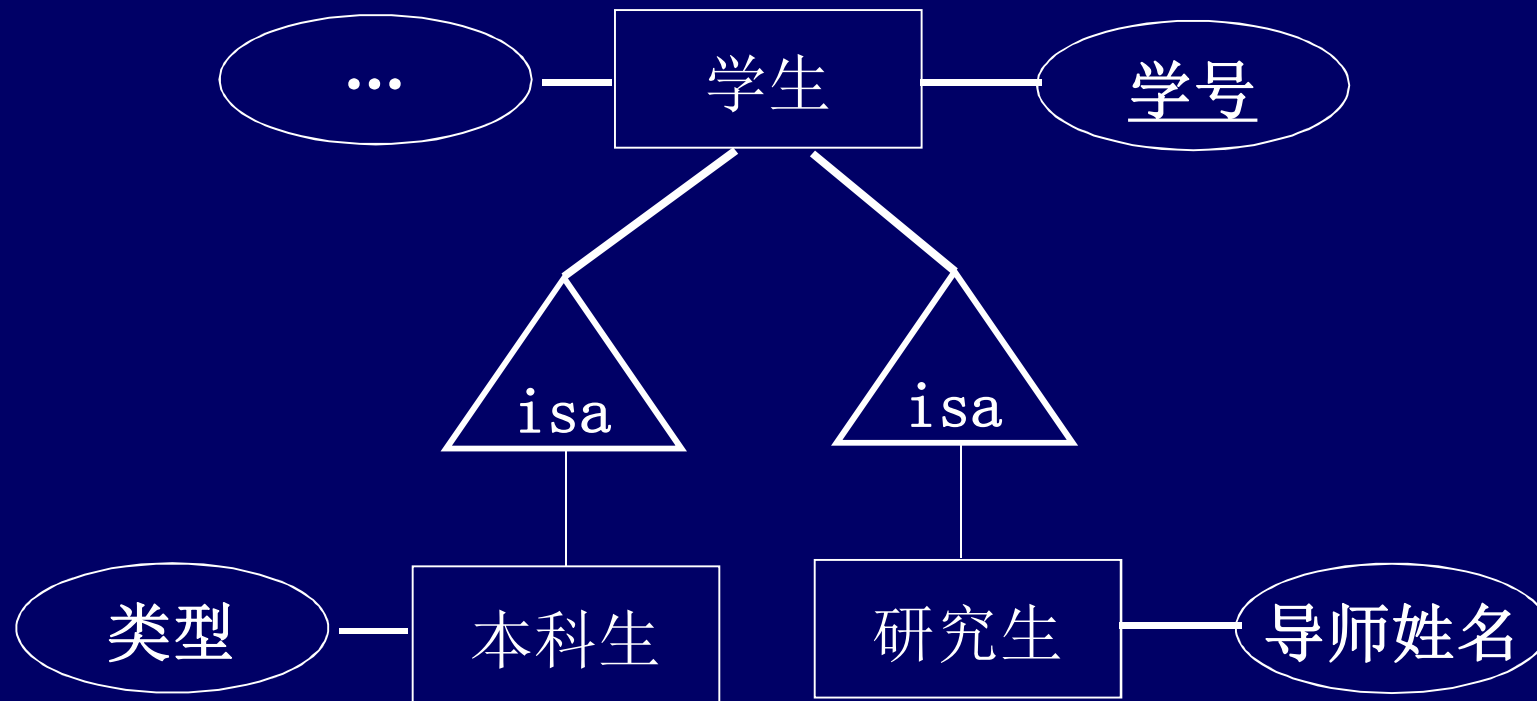
Method 1:

为父类实体集建立一个关系模式 为每个子类实体集分别建立一个关系模式.



Method 2:

如果子类实体集是不相交且全部的。

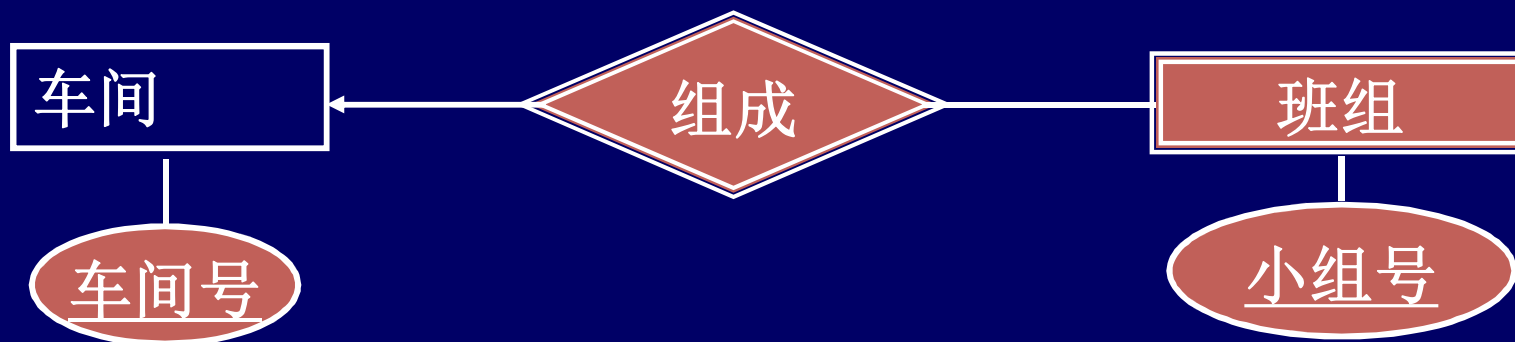


本科生(学号, 姓名, 类型 . . .)

研究生(学号, 姓名, 导师姓名 . . .)

* 弱实体集的变换

弱实体集A转化成一个关系模式，该关系模式的属性包括包括：
A自己的属性，以及为A提供主码属性的强实体集的主码。



班组(车间号, 小组号, . . .)

弱实体集与其它实体集间的联系如何处理？

逻辑模型的设计还包括：规范化的评价与选择，以及各种约束设计和视图的设计等！

3 数据库设计实例

(一) 问题描述——“航空票务”信息管理系统

1. 实体及联系描述

一个旅客可以凭有效证件（如：身份证），预订或登录班机。

旅客信息包括：身份证号，姓名，联系电话。

班机信息包括：班机号，（飞行）日期，可用座位，等。

机场信息包括：（机场）符号，城市，区域。

旅客订票时，需要确定：班机号，（飞行）日期，起飞机场及时间，到达机场及时间，（机票）种类（经济仓、公务仓）。
旅客登机时，需要登录信息：座位号，行李号。

2. 数据库应用描述

(1) 数据库系统应用环境描述：

- 集中应用系统
- 12个售票窗口
- 系统采用C2级数据安全标准
- 售票窗口采用操作员名字、口令登录管理。

(2) 数据库需要管理旅客、班机、机场的数量分别为 10^6 、 10^4 和 10^2 数量级。

(3) 订票应用：数量 10^4

- 利用图形界面录入：所订机票的（班机）日期，起飞和到达机场，机票种类。
- 应用系统显示：满足上述条件的班机、机场信息，包括：班机号，（飞行）日期，起飞和到达机场及时间等。
- 旅客经过选择，把选定结果写入应用系统：
（旅客）身份证号，姓名，联系电话；班机号，
（飞行）日期；起飞和到达机场及时间；客票种类，等。然后打印机票。

注：显然，一名旅客可能多次订购机票。

(4) 登录应用：数量 10^4

- 利用图形界面录入：身份证号；所订机票的班机号，（班机）日期。
- 应用系统显示：满足上述条件的旅客、班机信息。
- 经过操作员确认并指定座位号，把旅客及座位号，行李号）等写入应用系统。

注：显然，一名旅客可能登录不同班机。

(5) 起飞信息查询应用：数量 10^4

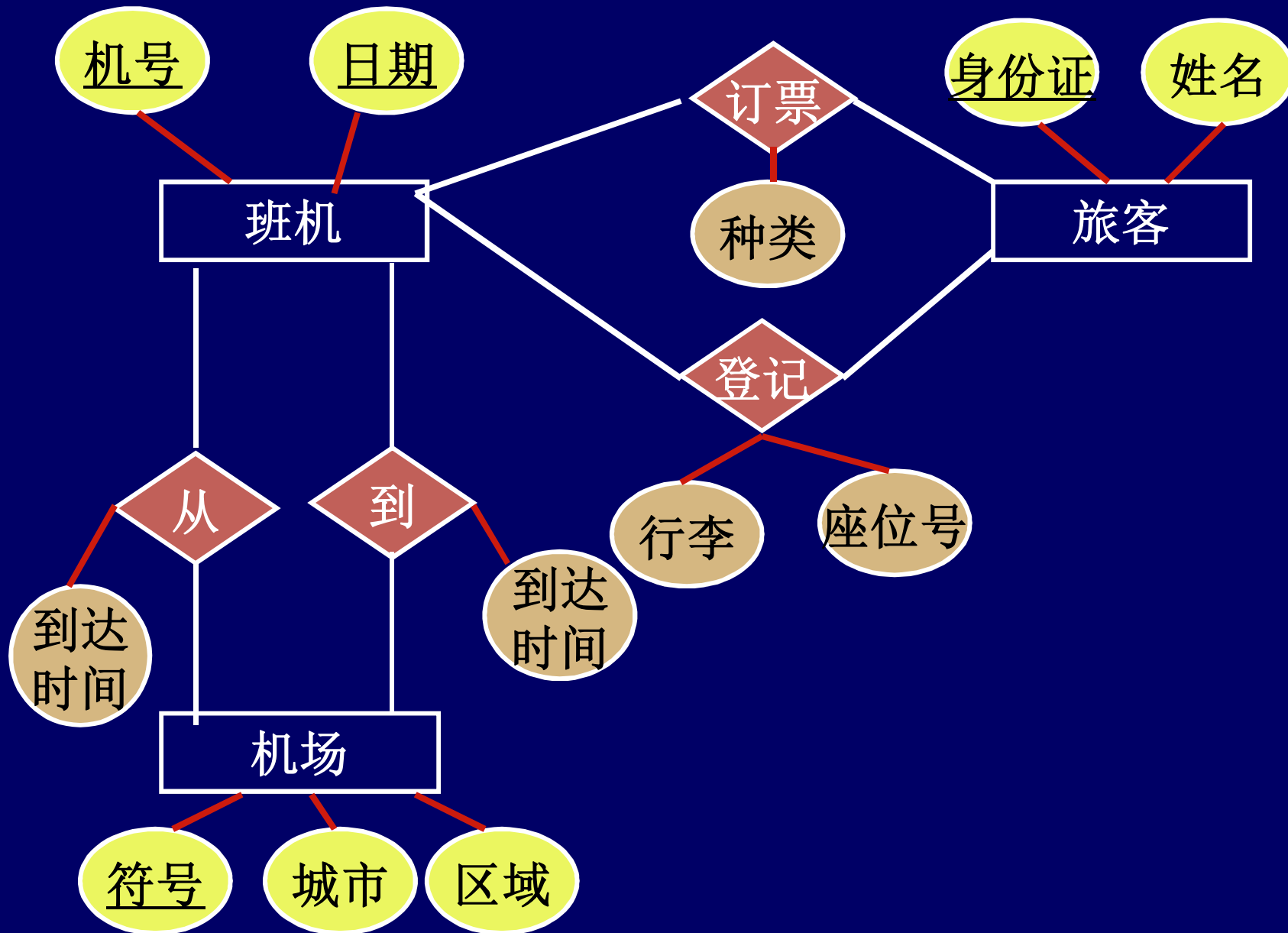
- 利用图形界面录入：（起飞）机场号；当前日期和时间。
- 应用系统显示：从当前日期和时间起，由指定（起飞）机场起飞的30架次班机，以及到达机场的信息。

(6) 系统数据录入、显示和更新

例如，班机和机场实例数据的录入、显示和更新。

注：因为一个航班可能经停多个机场，所以，一架次班机允许有多个起飞机场和到达机场。

(二) 概念设计——概念模型 (E-R图)



(三) 逻辑设计——逻辑模型（关系数据模型）

1. 关系模式设计

关注：

- 表名设计
- 列规范设计——列名与列类型设计

- 1) 机场（符号，区域，城市）
- 2) 班机（机号，日期，可用座位）
- 3) 旅客（身份证号，姓名，电话）
- 4) 从（机号，日期，符号，起飞时间）
- 5) 到（机号，日期，符号，到达时间）
- 6) 订票（身份证号，机号，日期，种类）
- 7) 登记（身份证号，机号，日期，座位号，行李）

2. 关系模式优化以及规范化与反规范化问题

优化：在不改变规范化程度上对关系模式进行合并

规范化与反规范化:根据应用需求决定数据的冗余程度!

“从(机号, 日期, 符号, 起飞时间)” 与
“到(机号, 日期, 符号, 到达时间)”

合并

从_到(机号, 日期, 起飞机场符号, 到达机场符号, 起飞时间, 到达时间)

3. 数据约束设计

(1) 键码约束设计 (实体完整性)

(2) 外码约束设计 (参照完整性)

机场 (符号, 区域, 城市)

从_到(机号, 日期, 起飞机场符号, , 到达机场符号起飞时间, 到达时间)

班机 (机号, 日期, 可用座位)

(3) 其它数据约束设计

- * 属性取值唯一性约束 (UNIQUE)
- * 空值约束 (NOT NULL) 和缺省值 (DEFAULT)
- * CHECK约束和断言约束

4. 子模式（视图）设计（略）

(四)数据库物理设计

1. 存储设计

(1) 表的存储空间估算

含：数据项（字段）、记录（元组）存储空间的估算。基础：数据类型对存储空间的需求。

(2) 数据库的存储空间估算

含：基本表（数据文件）、索引文件、日志文件、临时表、系统表存储空间的估算。

(3) 数据库块大小确定一块越大，性能越好？

(4) 确定表数据空间初始大小及增长率。

2. 索引设计

设计依据：数据库应用

- 例（1）对表“机场”分别创建关于“符号”的索引。
- （2）对表“班机”分别创建关于“机号”、“机号” + “日期”的索引。
- （3）对表“旅客”分别创建关于“身份证号”、“姓名”的索引。

3. RDBMS的选择

4. 日志空间和日志文件结构的选择

5. 备份数据空间设计以及备份策略的设计

五、数据库实施

1. 在选定的RDBMS下，创建数据库及其对象。
2. DBA根据数据库运行情况，进行数据库的性能调优工作