
管理系统中计算机应用

主讲老师：孟海鹰

第七章 系统设计

系统分析报告经过专家和用户评审、通过批准后，开发工作进入系统设计阶段。

设计阶段的目的

根据已经批准的系统分析报告，考虑实际的技术、经济和运行环境等条件，确定新系统的物理实施方案。

设计阶段的主要活动

- 1.系统总体设计，其中包括应用软件系统总体结构设计、数据库设计、计算机及网络系统配置方案设计。
- 2.系统详细设计，其中包括代码设计、用户界面设计、计算机处理过程设计。
- 3.编写系统设计报告。

主要内容

结构化设计的要求和主要图形工具的应用

模块处理过程的作用和基本设计方法

信息系统的主要操作界面及其设计原则和方法

代码的设计原则和类型和设计方法

数据库设计的步骤和方法

使用数据库管理系统创建和使用数据库、数据表

7.1 系统结构设计

7.1.1 总体结构设计任务

7.1.2 结构化设计的基本思想

7.1.3 模块分解的规则

7.1.4 控制结构图的绘制

7.1.1 总体结构设计任务

根据系统分析的逻辑模型设计应用软件系统的物理结构。

系统物理模型必须符合逻辑模型，能够完成逻辑模型所规定的信息处理功能，这是物理设计的基本要求。

系统应具有可修改性，即易读，易于查错、改错，可以根据环境和用户要求的变化进行各种改变和改进。

7.1.2 结构化设计的基本思想

基本理念：把大型的应用软件分解为多层的模块

核心内容：模块化、结构化、自顶向下逐步求精

(1) 程序结构化设计

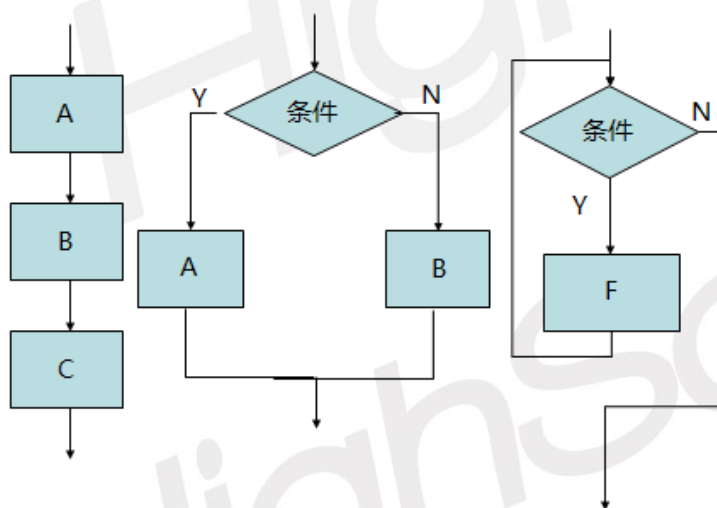
保持程序的一致性、可读性和可维护性

三种基本程序结构：顺序、选择、循环

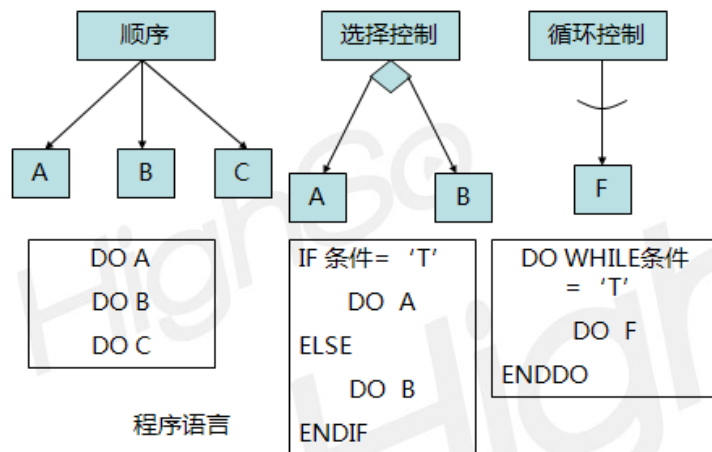
(2) 模块的结构化设计

(3) 控制结构图

程序流程图



模块结构图

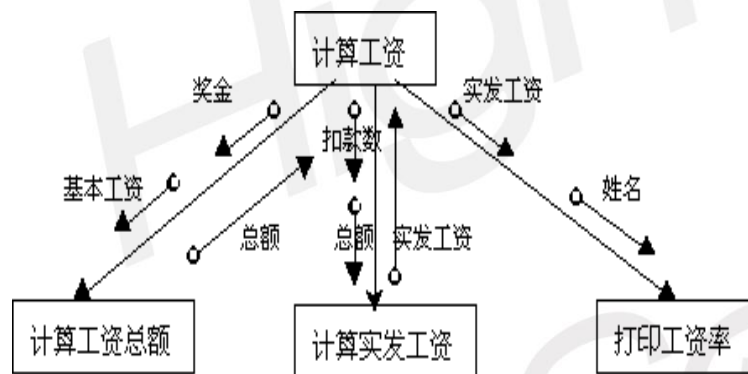


控制结构图

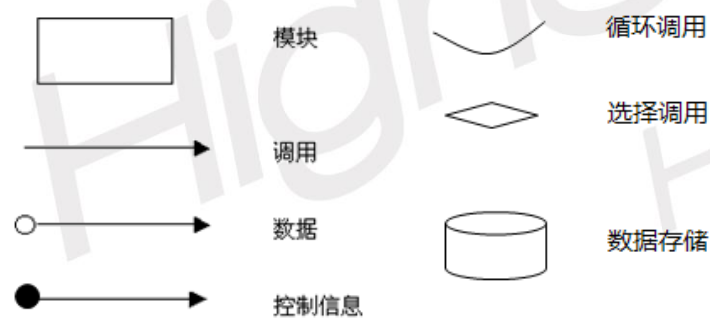
也称为软件结构图或模块结构图，它表示出一个系统的层次分解关系、模块调用关系、模块之间数据流和控制信息流的传递关系，它是系统物理结构的主要工具。

控制结构图既可以反映系统整体结构，又能反映系统的细节，能准确反映各组成部分(各模块)及它们之间的联系。

控制结构图实例



控制结构图符号



控制结构图的规定

模块调用的规则：

- (1) 每个模块有自身的任务，只有接收到上级模块的调用命令时才能执行。
- (2) 模块之间的通信只限于其直接上、下级模块。
- (3) 若有某模块要与非直接上、下级的其他模块发生通信联系，必须通过其上级模块进行传递。
- (4) 模块调用顺序为自上而下。

7.1.3 模块分解的规则

1. 模块的划分

模块是组成目标系统逻辑模型和物理模型的基本单位

特点是可以组合、分解和更换

分解的规则：

- (1) 系统项目容易开发
- (2) 降低项目开发成本
- (3) 系统有较高的可靠性
- (4) 系统容易维护

结构化设计的要求：高内聚、低耦合，即一个模块只实现一个明确的功能及之间的联系要松散，对其他模块的依赖程度低。

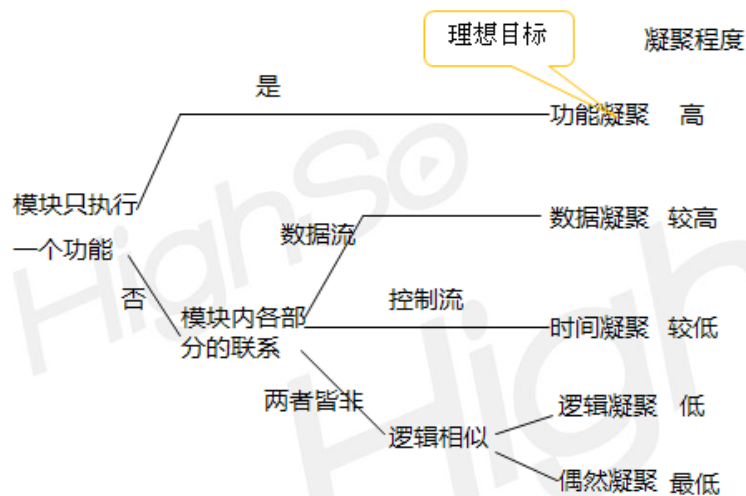
2. 模块凝聚

模块凝聚是用以衡量一个模块内部自身功能的内在联系是否紧密的指标，也是衡量模块质量好坏的重要指标。

模块按凝聚程度的高低可分为以下五级：

- (1) 偶然凝聚。
- (2) 逻辑凝聚。
- (3) 时间凝聚。
- (4) 数据凝聚。
- (5) 功能凝聚。

凝聚程度高，是结构化设计模块的理想目标。



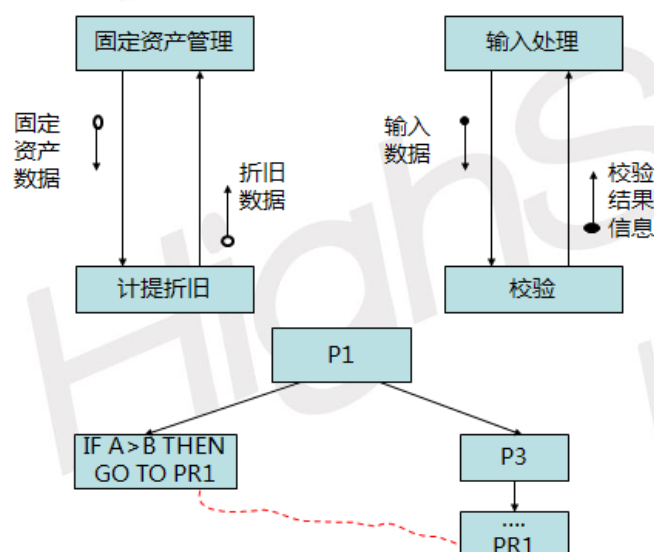
3. 模块耦合

模块间的信息联系方式，称为模块的耦合，它是衡量模块间结构性能的重要指标。

耦合有三种类型：

- (1) 数据耦合。
- (2) 控制耦合。
- (3) 非法耦合。

模块间数据耦合是最正常的方式，为保持模块的独立性，模块之间互相传递的数据要尽量少；要努力避免控制耦合，特别是避免自下而上传递控制信号；应消除任何形式的非法耦合。



7.14 控制结构图的绘制

绘制控制结构图的依据是数据流程图。

绘制控制结构图，首先是将上层数据流程图映射为上层控制结构图，由顶层数据流程图开始，逐级下推。

每一层数据流程图(DFD) 中的“处理功能”，映射为相应层次控制结构图中的“模块”；

而 DFD 中流入“处理功能”的数据流映射为输入模块的数据流。

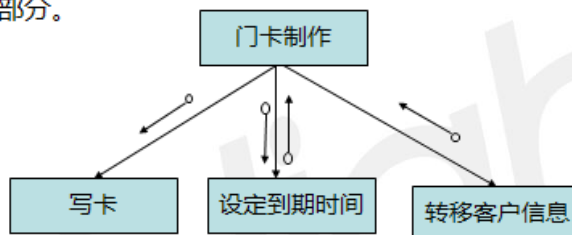
DFD 中流出“处理功能”的数据流映射成从“模块”中输出的数据流。

另一方面应按照模块分解的规则，将凝聚程度低的、或具有控制耦合、非法耦合的结构进行分解。

模块分解方法

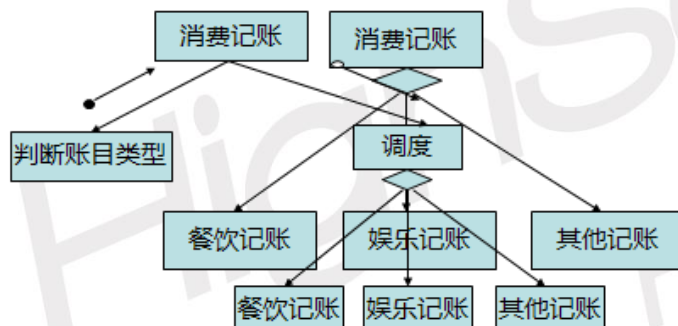
1.以转换为中心结构的分解

如果待分解的模块是一个数据凝聚的模块，即内部包含若干顺序执行且对某些数据进行转换处理，称为以转换为中心的结构。这种模块可分解为输入、处理、输出三大部分。



2.以业务为中心结构的分解

待分解的模块要处理几项逻辑上相似的业务，即它是一个逻辑凝聚的模块。这种模块可以将之分解为一个检查业务类型的模块和一个调度模块，根据不同的业务类型，调度模块调用不同的下层模块，进行不同的处理。



考核知识点与考核要求

识记：结构化设计的任务与方法、控制结构图、模块的凝聚性和耦合性

领会：模块、凝聚和耦合的概念、模块划分与软件开发成本的关系、高凝聚性和低耦合性的意义

简单应用：模块分解的规则与过程

综合应用：根据数据流程图和数据字典绘制控制结构图

例题

单项选择题：

1、系统总体设计不包括()

A . 软件系统总体结构设计

B . 数据库设计

C . 通信网络平台设计

D . 代码设计

2、结构化程序设计方法的基本程序结构包括()

A . 顺序结构

B . 选择结构

C . 循环结构

D . 顺序结构、选择结构和循环结构

3、一个模块内部各组成部分的处理彼此无关，偶然地组合在一起，这种凝聚是()

A . 逻辑凝聚

B . 偶然凝聚

C . 时间凝聚

D . 数据凝聚

名词解释题：

控制结构图

简答题：

简述模块间调用的规则。

7.2 处理过程设计

7.2.1 基本概念

7.2.2 用户口令处理过程的设计

7.2 处理过程设计

处理过程设计就是要对控制结构图中，每一个模块内部的处理过程，进行具体的描述。这种描述将成为以后编写处理程序的基础。

处理过程设计方法

在结构化系统设计方式中，处理程序的设计需要详细描述各种处理所用的算法和处理步骤等。

在面向对象的设计方法中，处理程序主要描述系统中各对象的方法及其对外部事件的响应和行为。

处理过程设计工具

详细地描述这个处理逻辑可以使用“输入—加工—输出”(Input Process Output, IPO)图。IPO 图将为编制程序提供指导，所以也叫做程序设计任务书。

IPO 图的主体是处理过程描述。

描述处理过程的工具，可以使用图形（程序流程图、N-S 图）、表格（决策表、决策树）和伪码（结构化语言）。

考核知识点与考核要求

识记：IPO 图、处理过程、处理工具

领会：结构化设计与面向对象设计对处理过程描述的异同

简单应用：使用决策树或决策表描述一个处理逻辑

例题

单项选择题：

1、IPO 图的主体是处理过程描述，描述处理过程的工具，可以使用()

A . 图形

B . 表格

-
- C . 伪码
 - D . 以上都是

2、在结构化设计方式中，处理程序的设计需要详细描述各种处理所用的()

- A . 算法
- B . 处理步骤
- C . 算法和处理步骤
- D . 对象

3、在面向对象的设计方法中，处理程序主要描述系统中()

- A . 各对象的方法
- B . 对象对外部事件的响应
- C . 对象对外部事件的行为
- D . 以上都是

名词解释题：

处理过程设计、IPO 图

7.3 界面设计

7.3.1 选择菜单的设计

7.3.2 各类窗口界面的设计

7.3.3 输入输出界面设计

界面设计是在总体结构约束下，对软件系统与用户之间交互接口的设计，包括对话窗口、选择菜单、输入输出窗口设计等很多方面。

7.3.1 选择菜单的设计

选择菜单是人机交互的主要界面之一，是用户操作使用软件系统最基本的工具。

如 WORD 的下拉菜单和快捷菜单。

设计好菜单的结构分组层次是创建菜单的最重要环节。

卡片分类法是一种探索人们如何将项目分组的技巧，在开发网站的结构上常用，可以提高用户在网站上搜索成功的可能性，目标是帮助人们找出项目分组的规律。

卡片分类法的步骤

卡片分类法会邀请用户参与选单的设计，并针对不同的场合采取灵活的工作方式。基本步骤如下：

- 1.需要分类的项目名称分别写在卡片上，打乱顺序摊开，让参与者看清；
- 2.请不同的参与者以自己认为合理的方式对卡片进行分组和归类；
- 3.要求参与者自行梳理各个分组，必要时为分组作出新的命名，写在空白卡片上；
- 4.当所有参与者完成后，根据最大共识的分类结果建立选单结构。

利用这种方法可以有效纠正技术人员先入为主、而最终用户难以使用的结构。

7.3.2 各类窗口界面的设计

应用系统实际就是由大量窗口界面连接而成。好的窗口设计特点：

- 1.对功能和信息的表现力强；
- 2.保持统一的风格；
- 3.突出核心功能。

人机对话窗口和信息提示窗口，是比较简单的窗口界面；但也是系统和用户及时交流的而重要手段。设计时要贯彻“用户友好”原则，顾及用户的接受能力和应用场景。

7.3.3 输入输出界面设计

输入界面设计原则：

- (1) 保证输入的正确性；
- (2) 输入数据的完整性；
- (3) 数据输入的效率。

输出界面设计：系统输出与信息系统开发的最终目的直接相关，是系统使用性能最直接的反映。设计需要考虑：

输出内容； 输出方式； 输出版面布局

考核知识点与考核要求

识记：菜单、卡片分类法、人机对话窗口、信息提示窗口

领会：界面的用户友好型、常用的设计原则和方法、输入界面的设计原则、输出界面设计的要求

简单应用：应用系统操作界面的评价

综合应用：用卡片分类法设计菜单

例题

单项选择题：

1、下列关于卡片分类法的说法中，错误的是()

- A . 在系统选单的设计中，有的小项目需要整理和归类，可以借用卡片分类法
- B . 卡片分类法不会邀请用户参与选单的设计过程
- C . 卡片分类法会针对不同的场合采取灵活的工作方式
- D . 用卡片分类法可以纠正技术人员先入为主、而最终用户难以查询使用的选单结构

2、人机对话窗口和信息提示窗口的设计，要注意不要()

- A . 贯彻“用户友好”原则
- B . 增强对用户的服务意识
- C . 避免直陈“你犯了某类错误”等负面用语
- D . 多用行话及专业术语

名词解释题：

卡片分类法

简答题：

简述设计输入界面应注意的原则。

名词解释题：

卡片分类法

简答题：

简述设计输入界面应注意的原则。

7.4.1 代码设计方法

7.4.2 代码的类型

7.4.3 代码的校验

7.4.1 代码设计方法

代码也叫信息编码，是作为事物（实体）唯一标识的、一组有序字符组合。必须便于计算机识别，处理，有利于人们使用。

代码是计算机和人都容易理解的符号(或语言)，能够实现人和计算机的沟通。

规范有序的编码是各个信息系统运作和彼此联系的基础。

代码设计是用规范化的编码符号来标识所有信息实体的工作过程。

代码的重要性

- 1.可以唯一地标识一个分类对象(实体)；
- 2.加快输入，减少出错，提高处理效率；
- 3.便于存储和检索，节省存储空间；
- 4.使数据的表达标准化，简化处理程序；
- 5.为全局数据一致性提供了基础。

代码设计的原则

- 1.唯一性：代码是区别系统中每个实体或属性的唯一标识。
- 2.简单性
- 3.可识别性
- 4.可扩充性
- 5.合理性
- 6.规范性

7.4.2 代码的类型

代码符号的表示形式。

(1) 顺序码。也叫序列码，用连续数字作为每个实体的标识。编码顺序可以是实体出现的先后，或实体名的字母顺序等。

优点是简单、易处理、易扩充、用途广；

缺点是没有逻辑含义、不能表示信息特征、无法插入、删除数据将造成空码。

(2) 重复码。采用与原来手工系统相同的编码。

优点是容易被原系统人员接受、易实现、便于推广；

缺点是不能任意更改、可能不尽合理。

(3) 成组码。也叫位别码、数字码、分组码，是最常用的编码之一。将代码分为几段(组)，每段表示一种含义，每段都由连续数字组成。

优点是简单、方便、能够反映出分类体系、易校对、易处理；

缺点是位数多不便记忆，必须为每段预留编码，否则不易扩充。

(4) 表意码。也称字符码或助记码，将表示实体特征的文字、数字或记号直接作为编码。

优点是可直接明白编码含义、易理解、易记忆；

缺点是编码长度位数可变，给分类、处理带来不便。

7.4.3 代码的校验

增加一位校验位

ASCII 码最高位为 0

考核知识点与考核要求

识记：代码的重要性、代码的含义、代码的类型

领会：代码设计的原则、代码校验技术

简单应用：用实例编制代码

例题

单项选择题：

1、为了唯一地、正确无误地标识系统中的每个实体，必须为管理信息系统建立相应的()

A . 密码系统

B . 代号系统

C . 管理系统

D . 代码系统

2、有些代码需要面向最终用户，如身份证号，这类代码的设计需要有()

A . 有较好的可识别性

B . 便于记忆和区分

C . 减少输入错误的机会

D . 以上都是

名词解释题：

成组码

简答题：

1、简述代码设计的原则。

2、简述常见的代码类型。

7.5 数据库的设计

7.5.1 要求和步骤

7.5.2 概念结构设计

7.5.3 逻辑结构设计

7.5.1 要求和步骤

数据库设计的目标是建立一个合适的数据库模型。这个数据库模型应当是：

(1)满足用户要求：既能合理地组织用户需要的所有数据，又能支持用户对数据的所有处理功能。

(2)满足某个数据库管理系统的要求：能够在数据库管理系统中实现。

(3)具有较高的范式：数据完整性好、效益高，便于理解和维护，没有数据冲突。

数据库设计步骤

数据库设计可以分为概念结构设计、逻辑结构设计和物理结构设计三个阶段。

(1)概念结构设计。数据库设计的第一个阶段。利用在分析阶段得到的系统的数据流程图和数据字典，结合数据规范化的理论，用概念数据模型将用户的数据需求明确地表示出来。

概念数据模型是面向问题的模型，反映了用户的现实工作环境，是与数据库的具体实现技术无关的。不具备计算机知识和数据库知识的管理人员容易理解，同时又很容易向逻辑数据模型转换。建立系统概念数据模型的过程叫做概念结构设计。

(2)逻辑结构设计。根据已经建立的概念数据模型，以及所采用的某个数据库管理系统软件的数据模型特性，按照一定的转换规则，把概念模型转换为这个数据库管理系统所能够接受的逻辑数据模型。

逻辑数据模型是用户通过数据库管理系统看到的现实世界，描述了数据库数据的整体结构。

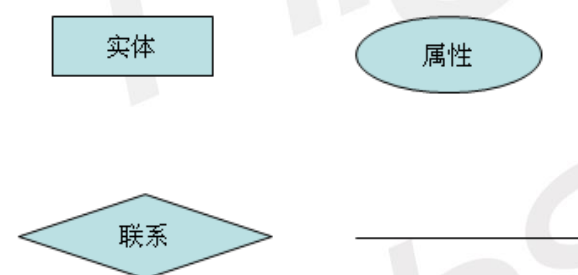
(3)物理结构设计。为一个确定的逻辑数据模型选择一个最适合应用要求的物理结构的过程，就叫做数据库的物理结构设计。数据库在物理设备上的存储结构和存取方法称为数据库的物理数据模型，不但受 DBMS 控制，而且与计算机软硬件相关。

7.5.2 概念结构设计

描述概念数据模型的主要工具是 E-R(实体—联系) 模型，或者叫做 E-R 图。利用 E-R 图实现概念结构设计的方法就叫做 E-R 方法。

概念模型的表示方法

E-R 图主要是由实体、属性和联系三个要素构成的。在 E-R 图中，使用了下面四种基本的图形符号。



1.确定系统实体、属性及联系

利用系统分析阶段建立的数据字典，并对照数据流程图对系统中的各个数据项进行分类、组织，确定系统中的实体、实体的属性、标识实体的码以及实体之间联系的类型。

2.确定局部(分) E-R 图

在数据字典中“数据项”是基本数据单位，一般可以作为实体的属性。作为属性必须是不可再分的数据项，也就是说在属性中不能包含其他的属性。“数据结构”、“数据存储”和“数据流”条目都可以作为实体，因为它们总是包含了若干的数据项。

在这些实体中有下划线的属性可以作为实体的码，实体之间存在着 1:1、1:n 和 m:n 几种联系。

根据上面的分析，可以画出部分实体 - 联系图。

3.集成完整(总) E-R 图

公共实体合并

在集成时应当注意如下几点：

- (1) 消除不必要的冗余实体、属性和联系。
- (2) 解决各分 E-R 图之间的冲突。
- (3) 根据情况修改或重构 E-R 图。

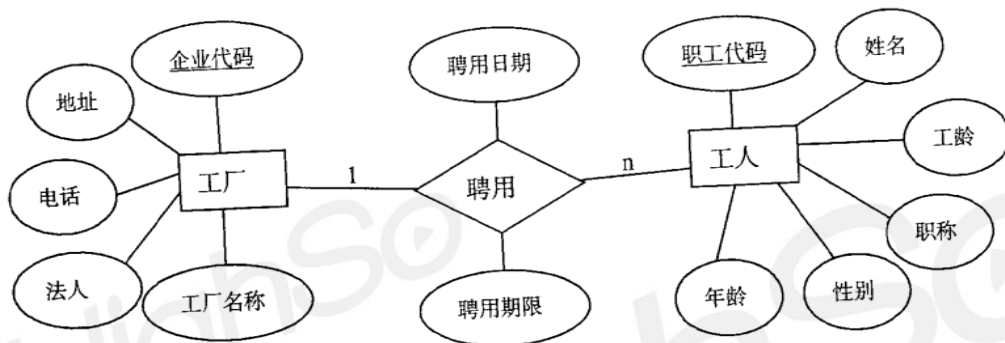
7.5.3 逻辑结构设计

逻辑结构设计任务，就是把概念结构设计阶段建立的基本 E-R 图，按选定的管理系统软件支持的数据模型(层次、网状、关系)，转换成相应的逻辑模型。

E-R 图转换关系模型的原则

- 1.一个实体转换为一个关系，实体的属性就是关系的属性，实体的码就是关系的码。
- 2.一个联系也转换为一个关系，联系的属性及联系所连接的实体的码都转换为关系的属性，但是关系的码会根据联系的类型变化，如果是：
 - ①1:1 联系，两端实体的码都成为关系的候选码。
 - ②1:n 联系，n 端实体的码成为关系的码。
 - ③m:n 联系，两端实体码的组合成为关系的码。
- 3.具有相同码的关系可以合并。

例题：按照 E-R 图向关系模型转换的原则将下列 E-R 图转换为关系。



考核知识点与考核要求

识记：设计要求、设计步骤、概念模型、逻辑模型

领会：概念模型(E-R 模型)的设计依据、E-R 模型向关系模型转换的原则

简单应用：概念模型的设计过程（数据模型到 E-R 模型）

综合应用：逻辑模型的设计过程（E-R 模型到关系模型）

例题

单项选择题：

1、数据库设计的目标是建立一个合适的数据模型。这个数据模型要求()

- A . 满足用户要求
- B . 满足数据库管理系统要求
- C . 具有较高的范式
- D . 以上都是

2、SQL SERVER 2000 支持的数据模型是()

- A . 层次模型
- B . 网状模型
- C . 关系模型
- D . 以上都不是

名词解释题：

概念结构设计

简答题：

E-R 图向关系模型的转换

应用题

1、某建筑单位可以承接各种工程。若每项工程均有一位负责人，每位负责人只能承包一项工程。每项工程都要用多种建材，每种建材又可以用于不同的工程。若每种建材只由一位供应商提供，每位供应商可以提供多种建材。

假定供应商提供的建材数量为 PMQ，各工程需要的建材数为 SMQ，每位承包人的工期为 SRT。如果已知 4 个实体如下：

工程：编号，名称，承包价，合同号 S(SN, ST, SP, sc)

建材：编号，名称，计量单位，单价 M(MN, MT, MU, MP)

供应商：编号，名称，地址，电话，传真 P(PN, PT, PAD, PTEL, PFX)

承包人：编号，姓名，职称，工作单位 R(RN, RM, RF, RU)

请先画出完整的 E—R 图，再转换成优化的关系模型。

7.6 数据库的物理实现

7.6.1 数据库的创建

7.6.2 数据库的操作

7.6.3 数据表的创建

7.6.4 数据表的操作

7.6.5 系统设计说明书

7.6.1 数据库的创建

使用 SQL Server 2000 创建

数据库文件：主数据文件 MDF、辅助数据文件 NDF、事务日志文件 LDF

用向导创建：企业管理器中逐层展开“控制台根目录”——指定“数据库”目录——“工具”选项——“向导”

用对话框窗口创建：企业管理器中逐层展开“控制台根目录”——指定“数据库”目录——快捷菜单“新建数据库”

使用 Visual Foxpro 创建

数据库文件：主文件 DBC、备注文件 DCT、索引文件 DCX

使用对话框窗口创建：基本方法，也是最常用的方法。文件—新建—数据库

使用命令方式创建：CREATE DATABASE [文件名 | ?]

7.6.2 数据库的操作

数据库的打开（进行任何操作前）

在 SQL Server 2000 中

系统启动自动打开所有 DB, 当前 DB:master

1. 企业管理器—控制台根目录—选择
2. 查询分析器—工具栏—数据库下拉列表选择
3. 命令：use 数据库名

在 Visual Foxpro 中

只能打开一个

1. 对话方式打开：文件-打开-文件类型-数据库 DBC
2. 命令方式：open database [文件名|?] [exclusive | shared] [nouupdate]

显示数据库信息

SQL Server 中，企业管理器，右击数据库名，属性

配置数据库

SQL Server 中，企业管理器，右击数据库名，属性，选项

关闭数据库

SQL Server 2000 中，不能关闭

Visual Foxpro 中，命令：close database [all]

数据库的删除

SQL Server 2000 中：

使用对话框：企业管理器--控制台—右击—删除

使用命令：drop database 数据库名

Visual Foxpro 中：

命令：delete database <数据库文件名 | ?> [deletetables | recycle]

7.6.3 数据表的创建

Visual Foxpro 中分为数据库表和自由表，可以相互转化，以文件形式存放在外存储器，主文件名 **DBF**，索引文件 **CDX**，备注文件 **DBT**

SQL Server 2000 中都是数据库表，可以指定数据库及所属用户，省略则以当前数据库为表的数据库创建

使用 **SQL Server 2000** 创建

1.企业管理器：控制台根目录—表—操作菜单/快捷菜单—新建表

2.查询分析器：命令

使用 **Visual Foxpro** 创建

1.对话框窗口：数据库菜单—新建表/数据库设计器—新建表

2.SQL 命令

7.6.4 数据表的操作

查看和修改表结构

SQL Server 2000 中：

企业管理器--控制台—表—操作菜单/快捷菜单—设计表/属性

Visual Foxpro 中：

打开数据表—显示菜单—表设计器

追加数据记录

SQL Server 2000 中：

企业管理器--控制台—表—操作菜单/快捷菜单—打开表—返回所有行—空行输入

输入空值：ctrl+0

标识列不允许修改

Visual Foxpro 中：

新建数据表时根据提示直接输入

打开表的浏览窗口 — 表菜单 — 追加新纪录

也可以连续追加 — 显示 — 追加方式

或成组追加 — 表 — 追加记录 — 追加来源

浏览与查询

SQL Server 2000 中：

企业管理器 -- 控制台 — 表 — 操作菜单 / 快捷菜单 — 打开表 — 返回所有行 / 返回首行 / 查询
(打开查询设计器)

Visual Foxpro 中：

打开表的浏览窗口 — 表菜单 — 转到记录

数据记录的删除与恢复

SQL Server 2000 中：

表的浏览窗口 — 左键单击左端行选择器 (shift 选择多行) -- delete

Visual Foxpro 中：

逻辑删除 (可恢复)

Set delete on | off ：是否显示带删除标记记录

Recall [范围|all] [<for 条件>] [<while 条件>]

物理删除 (不可恢复)

Pack ：物理删除带删除标记的记录

Zap ：物理删除全部记录

数据表的备份

SQL Server 2000 中：

企业管理器 -- 控制台 — 表 — 快捷菜单 — 所有任务 — 导出数据

Visual Foxpro 中：

文件—导出

数据表的索引

SQL Server 2000 中:

企业管理器—控制台—表—操作菜单/快捷菜单—所有任务—管理索引

Visual Foxpro 中 :

打开表设计器—索引选项

7.6.5 系统设计说明书

系统设计说明书又称为系统设计报告，是系统设计的最后成果，也是新系统的物理模型和系统实施的依据。

编写系统说明书的要求是：全面、清楚、准确、详细地阐明系统实施过程中的具体方法、技术、手段及环境要求。

系统设计说明书的内容

(1)控制结构图及每一模块的详细说明；

(2)数据库设计说明；

(3)计算机和网络系统配置说明；

(4)代码设计说明；

(5)用户界面设计说明；

(6)计算机处理过程说明；

(7)实施费用估计。

考核知识点与考核要求

识记：物理模型、物理数据库、主数据文件、辅助数据文件、事务日志文件、数据库主文件、数据库备注文件、数据库索引文件、企业管理器、查询分析器、向导、设计器、系统设计说明书的主要内容和编写要求

领会：SQL Server2000 数据库与 Visual FoxPro 数据库的异同；SQL Server2000 数据表与 Visual FoxPro 数据表的异同；系统设计说明书在系统开发过程中的作用

简单应用：使用向导及设计器创建用户数据库和数据表

综合应用：使用操作对话框窗口实现表的各种操作

例题

单项选择题：

- 1、在 Visual FoxPro 中，数据库文件包括数据主文件、数据库备注和()
 - A . 辅助数据主文件
 - B . 事务日志文件
 - C . 数据库索引文件
 - D . 以上都是
- 2、在 SQL Server 2000 中，启动系统就自动打开全部已有的数据库，并且将系统数据库 master 作为当前数据库。如果用户想要选择其他的数据库，则()
 - A . 通过企业管理器
 - B . 通过查询分析器
 - C . 通过命令
 - D . 以上都是

名词解释题：

主数据文件、系统设计说明书

简答题：

简述系统设计说明书的内容。

本章重点、难点

重点：

系统结构设计的原则

代码设计原则

概念模型的设计

逻辑模型的设计

创建数据表

操作使用数据表

难点：

模块的分解

绘制控制结构图

数据流程图到 E-R 图的转换

E-R 图到关系模型的转换

数据库和数据表的创建