

第7章

数据库需求分析与规划设计



本章内容

- 数据库需求分析与规划基本概念，数据库设计的要点
- 数据库设计的基本步骤和流程顺序
- 数据库设计的事实发现技术
- PowerDesigner 与数据库建模，包括概念模型图和物理模型图的产生，报告书的形成

7-1 数据库需求分析与规划基本概念



学习目标

- 了解数据库设计的要点。
- 掌握事实发现技术、基本步骤及完成数据库需求分析实例。

7-1-1 数据库设计的要点

1. 数据库设计的定义

数据库设计是指对于一个给定的应用环境，创建一个性能良好、能满足不同用户使用要求、又能被选定的 DBMS 所接受的数据库模式，建立数据库及其应用系统，使之能有效地存储数据，满足用户的信息要求和处理要求。

2. 数据库设计的主要内容

数据库设计内容包括静态设计，动态特性设计和物理设计。所谓静态设计是指结构特性设计，根据给定应用环境，设计数据库的数据模型或数据库模式，它包括概念结构设计和逻辑结构设计。概念结构设计的里程碑是从现实世界抽象出实体及其间的关系，理论上将 E-R 关系图作为设计的结论；而所谓的逻辑结构设计是指全局的关系逻辑结构设计，设计的里程碑将产生关系表、属性及其域定义、表的映射关系等，其中最为重要的是表的映射法则的确立。

所谓动态特性设计是指确定数据库用户的行为和动作，即数据库的行为特性设计，包括设计数据库查询、事务处理和报表处理等。动态的设计一般是在具体的应用项目中生成的，是根据项目对数据的局部视图的需求分析，而产生的视图、存储过程和触发器设计，重点是检验设计者 T-SQL 的

设计开发能力。

所谓物理设计是指根据动态特性，即应处理要求，在选定的 DBMS 环境下，把静态特性设计中得到的数据库模式加以物理实现，即设计数据库的存储模式和存取方法。如图 7-1 所示为数据库设计的主要内容及过程示意图。

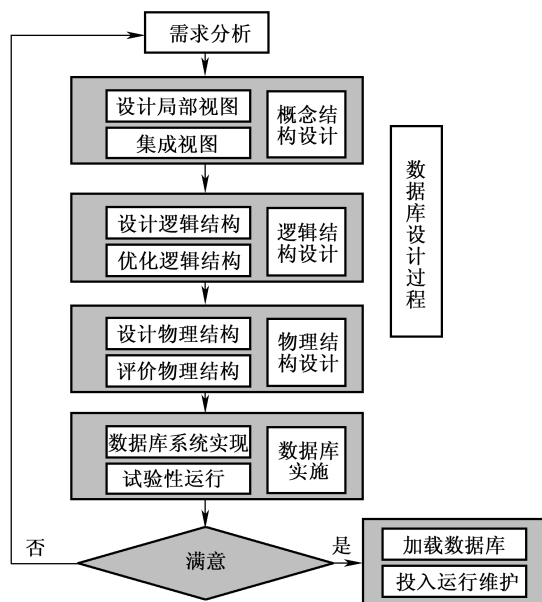


图 7-1 数据库设计的主要内容及过程示意图

3. 数据库设计的基本原则

根据数据库基本理论，数据库的设计是三分技术、七分管理和十二分的基础数据。我们学习 SQL Server 2005，SQL 及 T-SQL 是数据库的开发设计技术，是学习本门课程的基本功，但其主要应用在数据库的设计开发阶段，从软件项目的长远看仅是很短暂的一段时间。数据库一旦运营起来后，主要的工作则是 DBA 日常的数据库管理工作，该项工作将一直持续到软件项目的生命周期结束，是非常漫长的。而无论是设计开发还是管理运营维护数据库，底层的基础数据是最具有核心价值的，因此我们必须牢牢掌握住对基础数据的控制权。

数据库设计的基本原则可以归纳为以下几点：

- 1) 数据库的设计必须将硬件、软件和管理界面有机地结合在一起。
- 2) 数据库设计和应用设计要相互结合，要求达到减少数据冗余、实现数据共享的目的。
- 3) 数据库设计是结构设计和行为特性设计相结合的产物，但现实还没有有效工具使之能较好的结合，说到底就是软件的设计思想和数据库的设计思想并没有非常好的融合，这也为后期设计的无法统一埋下了问题的伏笔。
- 4) 数据库设计需要用户的参与以及具备发展的眼光。用户的参与很多时候流于表象，但是对于实际项目而言又绝非表象，这一矛盾需要数据库设计者具备很好的数据需求分析能力以及较好的情商，这一点不是单纯靠看几本书就可以理解的；而所谓发展的眼光是指系统不仅要满足用户目前的需求，也应满足近期要求，还要对于远期需求有相应的处理方案。当然这一预见能力也非一朝一夕之功，是必须经长时间的项目历练才可以达到的修为。

7-1-2 事实发现技术

1. 数据库设计为什么需要事实的发现

软件项目比较难的地方在于收集用户的真实需求，即需求分析报告和数据库分析报告很难编写；究其原因，除了专业和行业瓶颈外，用户很难按照程序员的思维或者 DBA 的想法告诉你想要的东西，可能是无法获取项目事实的真实背景。因此我们需要一套事实的发现技术，以解决在实际项目中遇到的需求分析瓶颈。

2. 事实发现的时机

在生命周期的早期，主要的目的就是发现信息系统所需要的实际业务流程以及真实的业务数据。事实发现的时机应当是在任何软件项目内容没有启动之前发生的，是在没有任何计算机技术参与的情况下发生的，如同与人闲聊的状态，知晓对方是干什么的以及怎样干的即可，说到底就是和甲方闲聊和沟通，从言语间揣度项目的事实。

3. 事实发现的具体内容

根据事实的发现，我们需要将沟通的成果迅速转化为数据库设计需要的基本元素，包括数据库的规划（实体对象有哪些、实体的属性有哪些、域是如何定义的、实体的关系是什么等）、系统的边界定义（必须完成什么、可做和不可做的是什么，以及完全没有必要设计的内容是哪些等）、需求的收集、签订合同、规范合同双方的责、权、利等。

4. 应当收集哪些事实

我们应当收集如表 7-1 所示的一些事实。

表 7-1 需求分析中应当收集的事实内容

开发阶段	捕获的数据	产生的文档
数据库规划	MIS 的目标和目的	可行性研究报告和项目开发计划书
系统定义	业务定义	数据要求说明书
需求收集和分析	用户视图要求和系统说明	需求说明书
数据库设计	逻辑数据库设计	数据库设计说明书
应用程序设计	界面以及功能模块的设计	概要设计说明书—模块开发说明
详细代码设计	程序算法设计	详细设计说明书
测试	数据的输入和输出设计	
管理与维护	DBMS 的管理和维护	

5. 事实发现技术的具体步骤

在一个软件信息系统的需求分析调查过程中，通常的事实发现技术包括以下几个内容：

1) 检查公司业务文档；查阅记录等。

在进行事实发现的时候，检查与目前系统相关的文档、表格、报告和文件是非常快速的解决办法，如表 7-2 所示。

表 7-2 文档检查列表

文档的用途	有用资源示例
描述数据库的问题和需求	内部备份录、会议记录、员工客户记录、业务过程记录
描述受问题影响的业务	组织图表、任务陈述、事务战略计划、任务工作描述
描述当前的系统	数据流程图和图表、用户培训手册

2) 面谈, 开调查会; 询问; 请专人介绍。

首先说明一点, 面谈是需要良好的沟通能力的, 即能够和具有不同价值观、喜好、个性、动机的人打交道, 也就是我们平时所说的情商较高的人。面谈的类型分为两种: 组织谈话和非组织谈话, 组织谈话效率比较高, 得到的信息较为准确, 一般以会议为主, 但需要提前准备好会议纪要以及会后的会议内容补充, 最好形成正式组织文档。非正式谈话比较自由, 但谈话的对象应该有所选择。面谈的优缺点如表 7-3 所示。

表 7-3 面谈的优缺点列表

优点	缺点
可以按照预先的内容进行交谈	浪费时间、代价昂贵
可以就每一个问题进行强化、重述	是否成功依赖于谈话人的交流技巧
可以观察谈话对象的肢体语言	
可以自由的、开放的回答问题	
可以充分了解部分组织关系	

3) 观察公司运行中的业务流程, 跟班作业; 观察业务流程的优缺点如表 7-4 所示。

表 7-4 观察业务流程优缺点列表

优点	缺点
可以检查事实与设计数据的有效性	行为异常
可以准确地把握正在做的业务流程	不自觉遗漏观察具体的任务
可以描述任务的物理环境相对低廉	主观修饰
可以做实际的工作测量	

4) 同行业软件的业务研究。

5) 问卷调查; 设计调查表要用户填写。

最后, 我们将数据库应用程序生存期每个阶段获得数据的分类和生成文档总结在表 7-5 之中, 请读者自行参照对应。

表 7-5 数据库应用程序生存期阶段收集数据及文档列表

数据库应用程序生存期阶段	收集的数据	生成的文档
数据库规划	数据库项目的目标和任务	数据库应用程序的任务陈述和任务目标
系统定义	描述主要用户视图	定义数据库应用程序的分和边界; 定义要支持的用户视图

续表

数据库应用程序生存期阶段	收集的数据	生成的文档
需求收集和分析	用户视图和系统需求	用户和系统需求说明书
数据库设计	验证逻辑数据库设计的用户反映；目标 DBMS 提供的功能	概念/逻辑数据库设计；物理数据库设计
应用程序设计	用户对界面设计的反映	应用程序设计
DBMS 选择	目标 DBMS 提供的功能	DBMS 的评估和推荐
建立原型	用户对系统原型的反响	改进的用户需求和系统需求说明书
实现	目标系统提供的功能	
数据转换和加载	当前数据的格式；目标 DBMS 的数据导入性能	
测试	测试结果	采用的测试策略；测试结果分析
运行维护	性能测试结果；用户和系统需求的增加和变化	用户手册；性能分析；改变的用户需求和系统说明书

7-1-3 事实发现的基本步骤及实例

1. 第一步：数据库系统的任务分析

本步骤分为两个阶段，第一个阶段是明确组织结构和组织工作的基本流程，具体的方法是查阅组织的发展历程和历史，查阅组织的具体结构。第二个阶段是确认组织的目的和软件的目标，具体的方法是与组织首席领导进行正式谈话，其目的是取得组织领导的信任，明确软件的边界。

2. 第二步：确定数据库系统的任务目标

创建目标的过程应该包括与员工中的合适人选进行引导性的对话，自由对话通常是这个阶段最有用的，典型的问题如下：

- 1) 请描述您的工作。
- 2) 通常一天里面你会做些什么工作呢？
- 3) 你经常和哪些数据打交道？需要使用哪些类型的报告？格式我可以复印一下吗？
- 4) 公司给你的任务主要是什么？你是如何完成的？

此阶段我们调查的对象主要包括：公司的业务主管；业务经理、监理、助理；业务员。

3. 第三步：系统的边界定义

系统边界定义的目的是确定数据库的应用范围和边界以及它的主要用户的视图，一个特定类型的数据库应用视图必须支持一个特定的工作角色或者是业务范围。系统边界定义的步骤包括两点：

- 1) 定义数据库的系统边界。
- 2) 确定数据库系统的主要用户视图。

4. 第四步：需求的收集和分析

经过系统边界的定义，我们明确了什么任务必须完成，什么任务可完成可不完等，由于边界定义的完成，才使得我们可以产生更多的细节，最终产生用户的需求说明。需求的收集和分析包括下列三项内容：

- 1) 收集数据库系统的用户视图相关的更多信息

2) 收集数据库系统的系统需求信息

3) 管理数据库系统的用户视图

除此以外, 为了收集数据库系统用户视图更多的信息, 我们还需要在询问中关注以下的问题:

- 1) 数据库中具体的表应该有哪些具体类型的信息?
- 2) 特定用户经常进行什么样的操作?
- 3) 什么事务对于当前的业务操作非常重要?
- 4) 什么时候应该进行严格的事务运行机制?
- 5) 数据库的高峰期、正常期和低谷期一般是何时?
- 6) 数据库需要哪种类型的安全机制和数据库存储机制?
- 7) 是否存在对于用户非常敏感的数据?
- 8) 哪些数据需要经常做备份? 需要保存哪些历史数据?
- 9) 对于数据库的网络和共享有哪些要求?



案例学习: 完成学生管理系统的事实发现

某学校计算机系准备设计学生管理系统的软件, 以解决日益复杂的学生管理问题。本案例目的是实施数据库的规划设计; 按照事实发现的步骤, 应该如何进行呢?

1. 明确组织结构和组织工作的基本流程

该阶段方法是查阅组织的发展历程和历史, 查阅组织的具体结构。根据现在学生管理的基本情况, 我们大致可以得到如图 7-2 所示的组织结构图。

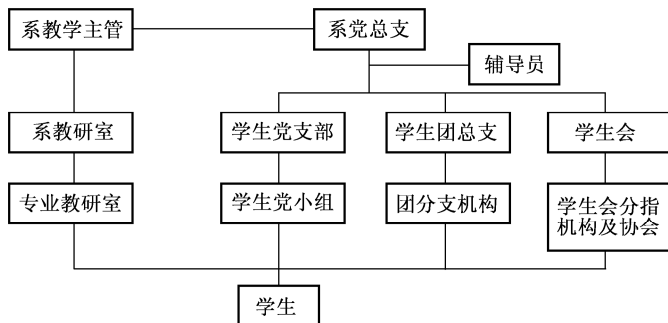


图 7-2 学生管理系统组织结构图

2. 确认组织的目的和软件的目标

该阶段的方法是与组织首席领导进行正式谈话, 其目的是取得组织领导的信任, 明确软件的边界。因此我们选择的谈话对象以及谈话基本内容包括:

谈话对象: 系总支书记

谈话的具体内容包括:

- 您现在设立这个软件项目的主要目的是什么?
- 有哪些数据让您特别头疼? 或者最占用你的日常工作时间呢?
- 你希望软件可以帮助你解决哪些方面的问题?

通过谈话我们确定软件项目定义的任务目标是收集、存储和控制本系学生在管理过程中产生的相关数据, 支持面向不同用户的学生数据查询和数据操作工作。该软件的基本边界是: 局域网络环

境，学生个体以及学生组织管理。

3. 确定数据库系统的任务目标

该阶段的方法是与数据库系统中可能涉及到的人员进行引导性的对话，自由提问是这个阶段的方式。其目的和任务是：确定数据库的操作角色和视图。具体的谈话内容如下：

(1) 谈话对象：系总支书记

谈话的具体内容包括：

- 通常一天里面你会做哪些学生工作呢？
- 您需要数据库系统迅速告诉您哪些事情呢？
- 一些日常的基本数据您是怎样获得的呢？谁来完成这些具体的事情？

(2) 谈话对象：辅导员

谈话的具体内容包括：

- 请描述您每天的具体工作。
- 您经常和哪些数据打交道？需要使用哪些类型的报告？格式我可以复印一下吗？
- 系里面给你的任务主要是什么？您是如何完成的？

经过上述的基本对话，我们可以得到对于数据库系统的基本任务目标：

- 维护（录入、更新和删除）学生基本情况；
- 维护（录入、更新和删除）宿舍卫生检查基本情况；
- 维护（录入、更新和删除）学生第二课堂情况；
- 维护（录入、更新和删除）学生上课情况信息；
- 维护（录入、更新和删除）成绩信息；
- 维护（录入、更新和删除）组织发展信息；
- 维护（录入、更新和删除）班级活动情况信息；
- 维护（录入、更新和删除）学生综合量化信息；
- 实现对以下学生信息的查询：学生基本信息、班级基本情况、学生违纪情况、学生成绩情况、组织发展情况、学生上课情况、学生综合量化情况。

4. 系统的边界定义

经过分析我们发现学生管理数据库的系统边界定义如图 7-3 所示，结合步骤 3 分析的结论，我们必须完成的系统包括：学生（教师，课程，教室，班级，党小组，学生会）基本数据管理模块，学生考勤管理模块，学生组织管理模块，学生第二课堂管理模块，学生成绩录入管理模块，学生综合量化管理模块；不应当涉及的模块包括：成绩管理系统，排课系统，行政办公管理系统等。

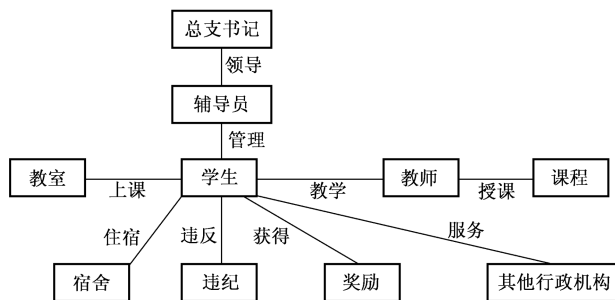


图 7-3 学生管理数据库的系统边界图

5. 总结数据库用户视图

经过调研,我们最终得到以下的数据库用户视图:

(1) 系总支书记

- 查询所有的班级信息。
- 查询所有的学生个人信息。
- 查询所有的成绩信息。
- 查询所有的考勤、宿舍卫生、奖励和惩罚。
- 查询所有的学生组织培养信息。
- 查询各种统计数据和量化考核数据。

(2) 辅导员

- 检索、维护(录入、更新和删除)给定班级的数据。
- 检索、维护(录入、更新和删除)给定班级的学生数据。
- 检索、维护(录入、更新和删除)给定班级的学生考勤数据。
- 检索、维护(录入、更新和删除)给定班级的学生宿舍卫生、奖励和惩罚数据。

(3) 学生

- 检索个人考试、考勤、宿舍卫生、组织培养情况。
- 个人量化考核。
- 维护个人的登录信息、检索班级信息以及相关的统计信息。

(4) 普通用户

检索班级信息以及相关的统计信息。

6. 形成数据库系统的事实调查说明书

作为数据库系统的说明书,应该详细描述以下的具体内容(仅仅涉及到事实调查的情况,不应涉及数据库的理论设计):数据库初始化大小,数据库增长速度及日志文件增长,记录查找的类型和主要使用系统表,网络和数据共享需求,性能,安全性,备份和恢复,用户界面等。本次项目确定的说明书基本调研参数内容如下:

(1) 初始数据库的大小

- 我系在校学生 1800 人,分布于 52 个自然班级中;每个班级平均学生 35~45 人。
- 现有辅导员 8 人。
- 平均每个班级有学生干部 12 人。

(2) 数据库的增长速度

- 一般而言,每个学期将会有 5 名学生退学或者休学;每月的注册人数不会超过学生人数的上限。

(3) 记录类型和平均数量

- 查询班级情况,每天 25 次。
- 查询学生基本情况,每天 800 次。
- 查询学生相关情况,每天 1200 次。

(4) 网络和共享访问需求

- 辅导员办公室必须安全地与系总支书记办公室的数据库服务器相连。
- 系统必须支持同时 100 人在线访问。

(5) 性能

- 每天上班时间要求单个记录查询时间在 1 秒。
- 高峰期为 5 秒。

(6) 安全性

- 数据库必须有口令保护。
- 每个用户必须根据身份分配到一个特定的用户视图。
- 数据库访问权限，主要包括：总支书记、辅导员、学生、普通用户。

(7) 备份和恢复

数据库设定在每周六半夜 12 点进行自动备份。

7-2 PowerDesigner 与数据库建模



学习目标

- 了解 PowerDesigner 基本的特性和发展历程，其主要功能和应用范围以及主要模块；
- 掌握通过 PowerDesigner 建立概念数据模型，认识其中的实体对象的属性、值域以及关键字，特别掌握实体之间关系的设计方法；
- 掌握通过 PowerDesigner 建立物理数据模型，掌握配置 PDM 图的全过程；
- 掌握将物理模型导入到数据库应用软件中的基本技术；
- 掌握生成数据库报告的基本技术。

7-2-1 PowerDesigner 简介

1. PowerDesigner 的历史变迁

PowerDesigner 是 Sybase 公司推出的一个集成了 UML（统一建模语言）和数据建模的 CASE（计算机辅助软件工程）工具。它不仅可以用于系统设计和开发的不同阶段（即商业流程分析、对象分析、对象设计以及开发阶段），而且可以满足管理、系统设计、开发等相关人员的使用。它是业界第一个同时提供数据库设计开发和应用开发的建模软件。

从 1989 年到 2003 年的十几年间，PowerDesigner 也经历了翻天覆地的变化，从一个单一数据库设计工具转变为一个全面的数据库设计开发和应用开发软件，图 7-4 描述了其发展简史。

2. PowerDesigner 主要功能模块

PowerDesigner 主要包含 4 个模块，即业务处理模型（BPM）、概念数据模型（CDM）、物理数据模型（PDM）和面向对象模型（OOM）。这 4 个模块覆盖了软件开发生命周期的各个阶段，图 7-5 表明了各个模块的相互关系及其作用。

在数据库规划设计中，首先进行的是数据库需求分析，并完成数据库概要模型设计；系统分析人员可以利用 CDM（概念模型图）设计出数据库逻辑结构模型；然后进行系统的详细设计，利用 PDM（物理数据模型）完成数据库的详细设计，包括存储过程、触发器、视图和索引等。

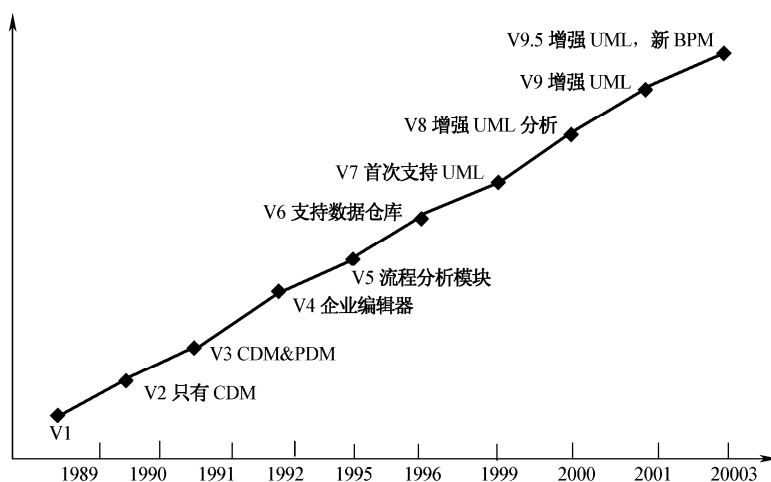


图 7-4 PowerDesigner 发展简史

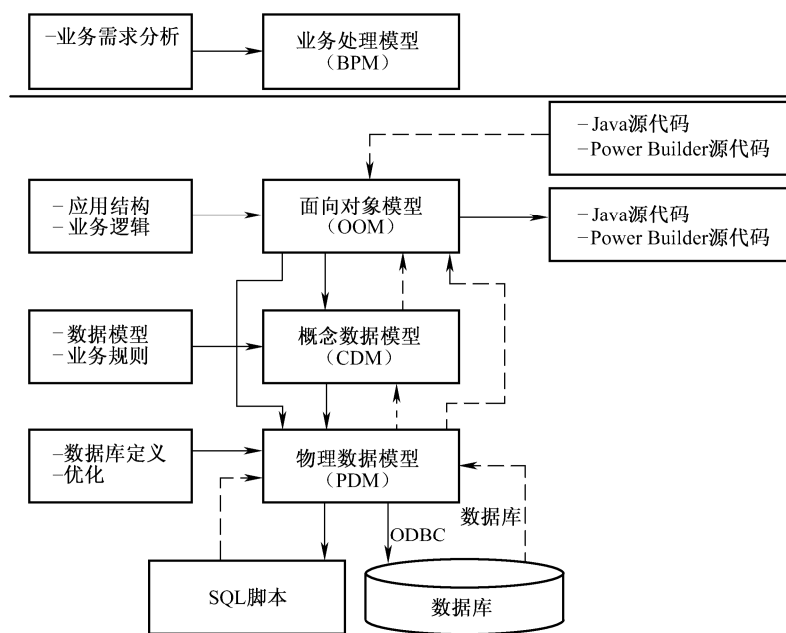


图 7-5 PowerDesigner 各个模块的相互关系和作用

3. PowerDesigner 的概念数据模型

概念数据模型 (Conceptual Data Model, CDM) 主要在系统开发的数据库设计阶段使用，是按用户的观点来对数据和信息进行建模，利用实体关系图 (E-R 图) 来实现。它描述系统中的各个实体以及相关实体之间的关系，是系统特性的静态描述。系统分析员通过 E-R 图来表达对系统静态特征的理解。E-R 图实际上相当于对系统的初步理解所形成的一个数据字典，系统的进一步开发将以此为基础。例如，描述学生与系之间关系的 E-R 图 (概念模型) 如图 7-6 所示。

该 E-R 图描述了学籍管理信息系统中需要处理的学生和系信息，以及它们之间所拥有的关系：

一名学生只能属于一个系，一个系可以有多名学生学习。它描述了系统的静态特征，即系统需要处理哪些基础数据，如何描述基础数据之间的关系，如何将这些基础数据有效地组织起来。下面是概念数据模型的主要功能：

- 1) 以图形化（E-R 图）的形式组织数据；
- 2) 检验数据设计的有效性和合理性；
- 3) 生成物理数据模型（PDM）；
- 4) 生成面向对象模型（OOM）；
- 5) 生成可定制模型报告。

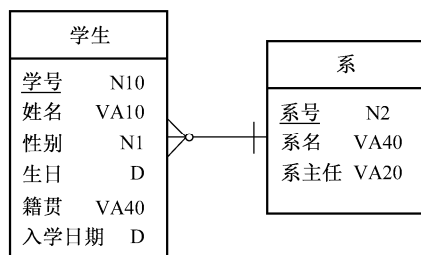


图 7-6 学生与系的 E-R 图

4. PowerDesigner 的物理数据模型

物理数据模型（Physical Data Model, PDM）提供了系统初始设计所需要的基础元素，以及相关元素之间的关系，但在数据库的物理设计阶段必须在此基础上进行详细的后台设计，包括数据库存储过程、触发器、视图和索引等。物理数据模型是以常用的 DBMS（数据库管理系统）理论为基础，将 CDM 中所建立的现实世界模型生成相应的 DBMS 的 SQL 语言脚本，利用该 SQL 脚本在数据库中产生现实世界信息的存储结构（表、约束等），并保证数据在数据库中的完整性和一致性。

利用概念数据模型可以自动生成物理数据模型，图 7-7 的物理数据模型就是通过上面的概念数据模型自动转换而成的，系统中包含两张表：学生表和系表，以及两张表之间所存在的主外键关系。

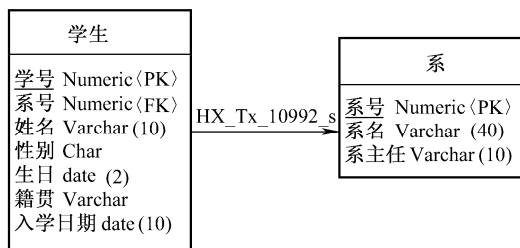


图 7-7 物理数据模型

下面是物理数据模型的主要功能：

- 1) 可以将数据库的物理设计结果从一种数据库移植到另一种数据库。
- 2) 可以利用逆向工程把已经存在的数据库物理结构重新生成物理模型或概念模型。
- 3) 可以生成可定制模型报告。
- 4) 可以转换为 OOM。
- 5) 完成多种数据库的详细物理设计。生成各种 DBMS（如 Oracle、Sybase、SQL Server 和 SQL

Anywhere 等 30 多种数据库) 的物理模型, 并生成数据库对象(如表、主键、外键等)的 SQL 语句脚本。

5. PowerDesigner 中 CDM (概念数据模型) 的操作控制台

PowerDesigner 展开后的操作控制台如图 7-8 所示, 该控制台可以用分层结构显示你的工作空间, 其中的输出窗口用于显示操作的结果, 结果列表用于显示生成、覆盖和检查模型结果, 以及设计环境的总体信息。图表窗口用于组织模型中的图表, 以图形方式显示模型中各对象之间的关系。

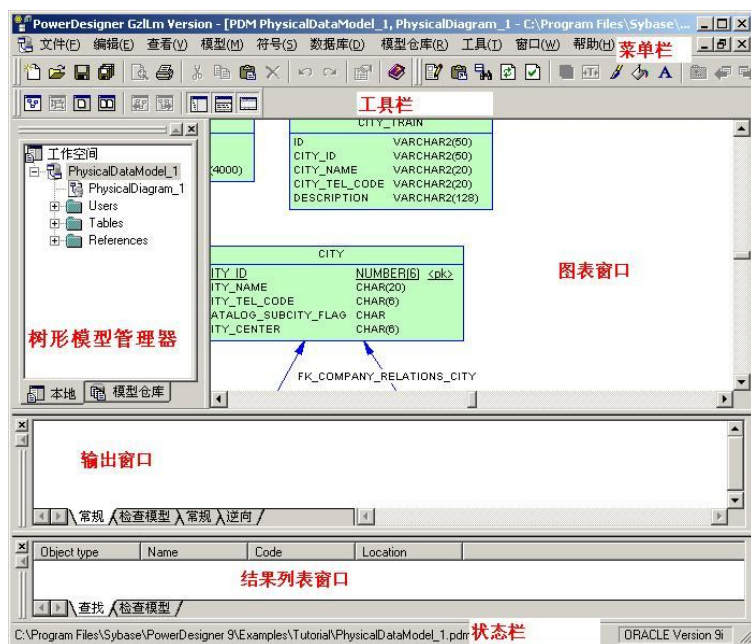


图 7-8 操作控制台说明

该操作控制台中最为重要的是工具栏(如图 7-9 所示), 表 7-6 详尽说明了工具栏中每个工具的作用。



图 7-9 工具栏

表 7-6 工具栏主要工具列表

图形	名称	操作	图形	名称	操作
	指针	选择符号		联合连接	插入联合连接符号
	整体选择	选择全部符号, 一起设置大小		文件	插入一个文件符号
	放大	放大视野范围		注释	插入注释符号
	缩小	缩小视野范围		连接, 扩展依赖	在图表中的符号之间画一个图形连接, 在注释和一个对象之间画一个注释连接, 在两个支持扩展依赖的对象间画一个扩展依赖

续表

图形	名称	操作	图形	名称	操作
	打开包图表	显示选择包的图表		主题	插入主题符号
	属性	显示选择的符号属性		文本	插入文本
	删除	删除符号		线条	插入一条线
	包	插入包符号		圆弧	插入一个圆弧
	实体	插入实体符号		长方形	插入一个长方形
	关系	插入关系符号		椭圆	插入一个椭圆
	继承	插入继承符号		圆角矩形	插入一个圆角矩形
	联合	插入联合符号		折线	插入一条折线

7-2-2 通过 PowerDesigner 建立概念数据模型（CDM）



案例学习：完成教务成绩管理系统的概念数据模型（CDM）设计

教务成绩管理系统的基本工作流程为：首先每学期期末考试前一段时间，由教务处管理人员登录系统后启动期末考试成绩录入系统，并进行教师基本信息的维护工作。而后各系辅导员登录系统，将本人所带的班级信息录入，并将该班级学生基本资料录入系统。在教务处规定的统一时段内（考试后一周内），教师根据教务处录入的教师资料登录系统，首先将本学期个人所带的课程信息录入（必须有相应的班级对应），而后根据选择的班级和课程，选择具体的学生，并录入该学生该课程的期末分数，分数分布按照各院校考试成绩表的规定执行。成绩录入完毕提交后，教师无法修改，只能由教务人员更正信息。对于录入后的成绩信息，学生、教师、教务、辅导人员都有权查看，但检索范围不同；同时系统应对成绩进行统计分析以及排名综合，并按照教务相关报表格式执行打印。

教务成绩管理系统定位为学历教育学生成绩管理，成绩为每学期末卷面成绩，系统设计操作对象概念范围包括教务管理人员，各系教学辅助人员，主考教师和参考学生，系统管理员。本软件项目开发系统定位在成绩管理，因此在项目分析中不要涉及以下软件外延概念：

学生管理；教师管理；课程管理；排课管理；考勤管理；学分管理；毕业成绩汇总管理；与其他应用软件系统的接口管理。

本软件项目开发系统的内涵是：仅仅涉及与成绩管理有关的适度信息管理，以及与成绩管理有关的信息对象适度的属性范围。

第一步：启动 PowerDesigner

本次设计我们使用的是 PowerDesigner 12.0 版本，启动后我们首先选择菜单“文件”→“新建”，在弹出的对话框中选择“模型类型”中的 Conceptual Data Model，即概念数据模型，命名 Model name 为“成绩管理概念模型”，如图 7-10 所示。

第二步：建立实体对象

在 CDM 操作界面，选择工具栏中的实体对象，然后在图表窗口中单击一次，建立一个实体对

象，代表成绩管理数据库中的学生对象。再双击该对象，在 General 选项卡设置 Name 为“学生表”（表示为显示出来的对象名称），Code 设置为“student”（表示最终生成的表对象名称），Comment 为对象描述，Number 为对象的序列号，Generate 表示将自动生成表对象，如图 7-11 所示。



图 7-10 新建概念数据模型



图 7-11 建立实体对象图

这里需要注意的是，Name 是属性的逻辑（显示）名称，而 Code 是属性的编码名称，最终将形成物理表的实际属性名。为了方便设计，一般将 Name 直接用中文说明，也就是实际编码名称的中文解释；而 code 应该设置成英文名称，一般尽可能用英文直译命名，以方便今后编码时调用的唯一性。

第三步：配置实体对象的属性、值域以及关键字（主键）

承接上一步，再选择 Attributes（属性）选项卡，在 Name 列（显示命名）中分别输入学生编号，学生姓名，学生性别，学生生日；在 Code 列（实际属性名称）中对应输入 sno，sname，sssex，sbirth。接下来开始配置 student 表的主键。

每一行属性中，都可以配置 MPD 选项。M（Mandatory）表示属性不可以为空；P（Primary Key）表示该属性为主键/键；D（Display）表示在界面中是否显示该属性。如果设定某属性为 P，则自然 M 选项将被勾选。在该项配置中，至少应配置每个表的主键 P（Primary Key）以及是否必填 M（Mandatory）。如本案例中，可用鼠标左键选中 sno 属性，勾选 P 项，则 M 自动勾选（实体完整性），对于需要必填的属性同样可以将 M 勾选，如姓名项，如图 7-12 所示。

输入属性后，要设置属性的数据类型。设置的方法是单击每个属性的 Data Type 单元格，在弹出的“Standard Data Types（标准数据类型）”对话框中进行配置。数据类型的配置可以简单地分为

数值类型配置和字符类型配置，对于数值类型无需配置长度大小，而字符类型则需要设置字符串类型的长度大小（length）。配置完毕后单击 OK 按钮确定。例如，设置“姓名”属性为 Variable Character 类型，长度为 30 位，见图 7-13。

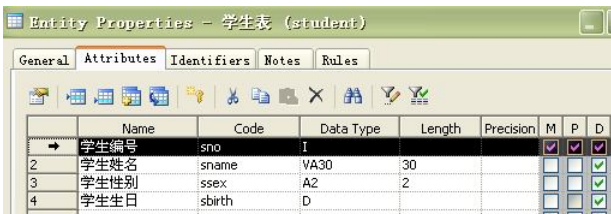


图 7-12 建立实体对象图

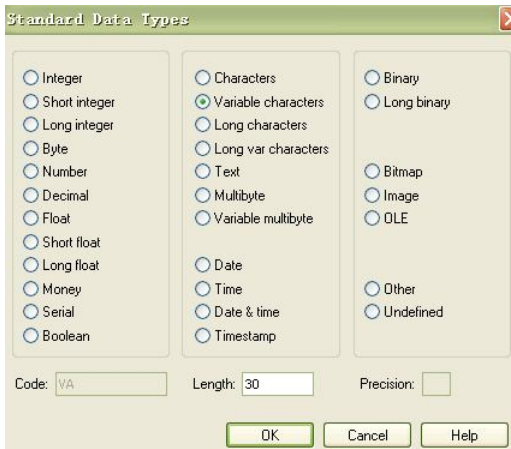


图 7-13 配置属性类型示意图

最后设定好的实体对象如图 7-14 所示。

<table><tr><td colspan="3">管理者</td></tr><tr><td>管理员编号</td><td><pi> 1</td><td><M></td></tr><tr><td>管理员姓名</td><td>VA20</td><td></td></tr><tr><td>管理员密码</td><td>VA20</td><td></td></tr><tr><td>等级</td><td>SI</td><td></td></tr><tr><td colspan="3">Identifier_1 <pi></td></tr></table>	管理者			管理员编号	<pi> 1	<M>	管理员姓名	VA20		管理员密码	VA20		等级	SI		Identifier_1 <pi>			<table><tr><td colspan="3">班级表</td></tr><tr><td>班级编号</td><td><pi> 1</td><td><M></td></tr><tr><td>班级名称</td><td>VA30</td><td></td></tr><tr><td>入学时间</td><td>D</td><td></td></tr><tr><td>班级人数</td><td>SI</td><td></td></tr><tr><td colspan="3">Identifier_1 <pi></td></tr></table>	班级表			班级编号	<pi> 1	<M>	班级名称	VA30		入学时间	D		班级人数	SI		Identifier_1 <pi>			<table><tr><td colspan="3">学生表</td></tr><tr><td>学生编号</td><td><pi> 1</td><td><M></td></tr><tr><td>学生姓名</td><td>VA30</td><td></td></tr><tr><td>学生性别</td><td>A2</td><td></td></tr><tr><td>学生生日</td><td>D</td><td></td></tr><tr><td colspan="3">Identifier_1 <pi></td></tr></table>	学生表			学生编号	<pi> 1	<M>	学生姓名	VA30		学生性别	A2		学生生日	D		Identifier_1 <pi>																				
管理者																																																																										
管理员编号	<pi> 1	<M>																																																																								
管理员姓名	VA20																																																																									
管理员密码	VA20																																																																									
等级	SI																																																																									
Identifier_1 <pi>																																																																										
班级表																																																																										
班级编号	<pi> 1	<M>																																																																								
班级名称	VA30																																																																									
入学时间	D																																																																									
班级人数	SI																																																																									
Identifier_1 <pi>																																																																										
学生表																																																																										
学生编号	<pi> 1	<M>																																																																								
学生姓名	VA30																																																																									
学生性别	A2																																																																									
学生生日	D																																																																									
Identifier_1 <pi>																																																																										
<table><tr><td colspan="3">教师表</td></tr><tr><td>教师编号</td><td><pi> 1</td><td><M></td></tr><tr><td>教师姓名</td><td>VA30</td><td></td></tr><tr><td>登录密码</td><td>VA20</td><td></td></tr><tr><td>职称</td><td>VA20</td><td></td></tr><tr><td>学生生日</td><td>D</td><td></td></tr><tr><td>学位</td><td>VA16</td><td></td></tr><tr><td>毕业院校</td><td>VA50</td><td></td></tr><tr><td>毕业专业</td><td>VA30</td><td></td></tr><tr><td>毕业日期</td><td>D</td><td></td></tr><tr><td>电子邮箱</td><td>VA30</td><td></td></tr><tr><td>联系电话</td><td>VA30</td><td></td></tr><tr><td colspan="3">Identifier_1 <pi></td></tr></table>	教师表			教师编号	<pi> 1	<M>	教师姓名	VA30		登录密码	VA20		职称	VA20		学生生日	D		学位	VA16		毕业院校	VA50		毕业专业	VA30		毕业日期	D		电子邮箱	VA30		联系电话	VA30		Identifier_1 <pi>			<table><tr><td colspan="3">系</td></tr><tr><td>系编号</td><td><pi> SI</td><td><M></td></tr><tr><td>系名称</td><td>VA20</td><td></td></tr><tr><td>联系电话</td><td>VA30</td><td></td></tr><tr><td>系主任</td><td>VA20</td><td></td></tr><tr><td colspan="3">Identifier_1 <pi></td></tr></table>	系			系编号	<pi> SI	<M>	系名称	VA20		联系电话	VA30		系主任	VA20		Identifier_1 <pi>			<table><tr><td colspan="3">课程表</td></tr><tr><td>课程编号</td><td><pi> 1</td><td><M></td></tr><tr><td>课程名称</td><td>VA40</td><td></td></tr><tr><td>课程性质</td><td>VA8</td><td></td></tr><tr><td colspan="3">Identifier_1 <pi></td></tr></table>	课程表			课程编号	<pi> 1	<M>	课程名称	VA40		课程性质	VA8		Identifier_1 <pi>		
教师表																																																																										
教师编号	<pi> 1	<M>																																																																								
教师姓名	VA30																																																																									
登录密码	VA20																																																																									
职称	VA20																																																																									
学生生日	D																																																																									
学位	VA16																																																																									
毕业院校	VA50																																																																									
毕业专业	VA30																																																																									
毕业日期	D																																																																									
电子邮箱	VA30																																																																									
联系电话	VA30																																																																									
Identifier_1 <pi>																																																																										
系																																																																										
系编号	<pi> SI	<M>																																																																								
系名称	VA20																																																																									
联系电话	VA30																																																																									
系主任	VA20																																																																									
Identifier_1 <pi>																																																																										
课程表																																																																										
课程编号	<pi> 1	<M>																																																																								
课程名称	VA40																																																																									
课程性质	VA8																																																																									
Identifier_1 <pi>																																																																										

图 7-14 成绩管理系统实体对象明细图

第四步：设置实体彼此的关系

选择工具栏中的关系工具（Relationship），在你认为有关系的两个实体之间“划”一下，则两个实体之间出现关系线，如图 7-15 所示。

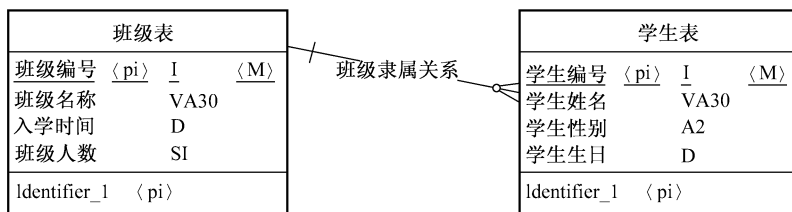


图 7-15 连接两个实体之间的关系

双击连接线，配置实体彼此之间的关系属性，在页式选项卡的 General 选项卡中定义关系名称（Name）和编码名称（Code），同上 Name 为显示名称（建议中文表示），Code 为实际存储名称（建议英文或拼音表示），如图 7-16 所示。

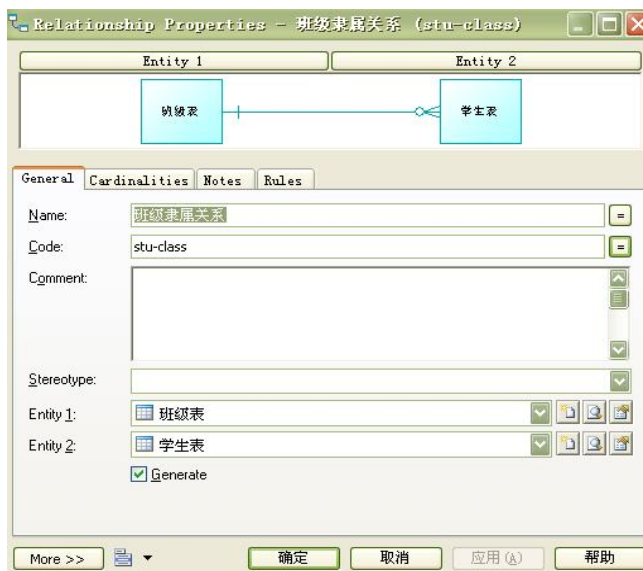


图 7-16 配置关系界面

仍然在该页式选项卡中，选择 Cardinalities（关系明细）选项卡，配置实体彼此之间的关系明细，如图 7-17 所示。

在图 7-17 第一行的配置中，可以看见 One-One、One-Many、Many-One、Many-Many 四个选项，即我们在理论中所熟悉的一对一关系、一对多关系、多对一关系、多对多关系。我们无需设置这一行的内容，因为只要配置好了下面的两行对应关系，则该行将自动变化。

下面两行分别需要设置“班级表 to 学生表”和“学生表 to 班级表”，为了具有普适性我们将这一关系简化为“X to Y”，其意义是：对于任何一个 X，与之对应的 Y 是一个还是多个。如“班级表 to 学生表”，我们可以配置为对于任何一个班级可以管理多个学生。具体的数值设置需要选择 Cardinality 下拉列表框，如图 7-18 所示。



图 7-17 配置关系明细界面

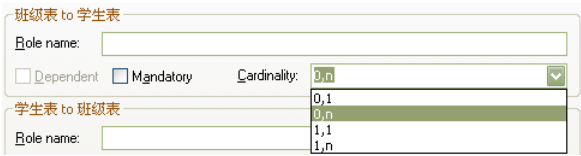


图 7-18 配置实体对应关系

其对应法则关系一共有四种，分别表示的意思是：

- 1) 0,n 至少 0 个，至多 n 个，逻辑表示为任意。
- 2) 0,1 至少 0 个，至多 1 个，逻辑表示为至多 1 个。
- 3) 1,1 至少 1 个，至多 1 个，逻辑表示为只有 1 个。
- 4) 1,n 至少 1 个，至多 n 个，逻辑表示为至少 1 个。

根据图 7-18 所示关系，对于“班级表 to 学生表”选择 Cardinality 值为 0,n，表示“对于一个班级，其管理的学生最少 0 个，最多 n 个”；反之，对于“学生表 to 班级表”选择 Cardinality 值为 1,n，表示“对于任意一个学生，管理他的班级至少 1 个，最多 1 个”。按照上面的方法配置完毕后，对应的关系自然变成 One-Many。同理，对于学生和班级之间的关系，可以按照图 7-19 所示设置，最后达到多对一的对应法则。

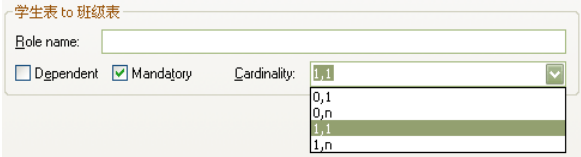


图 7-19 学生—班级对应关系

当完成实体关系设置工作后，即完成了概念模型的基本设置工作，如图 7-20 所示。

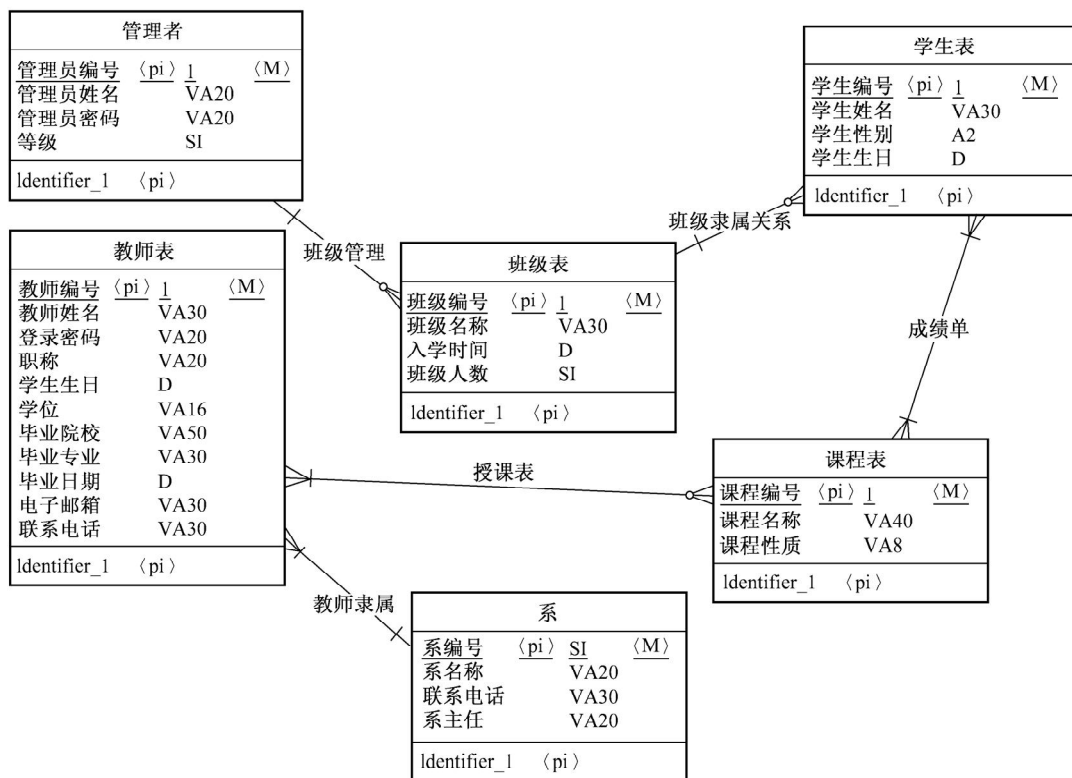


图 7-20 配置每个实体的对应关系

7-2-3 通过 PowerDesigner 建立物理数据模型 (PDM)

概念模型所关心的是现实世界的具体实体是什么,这些实体的属性及码是什么?这些实体彼此之间的关系是什么?而物理模型关心的是哪些实体或者关系形成了物理表,这些物理表彼此之间的依赖关系是什么?哪些属性被设置成了主键和外键等,毕竟物理模型的设计已经和实际的数据库设计软件很接近了。



案例学习: 完成教务成绩管理系统的物理数据模型 (PDM) 设计

PDM (Physical Data Model) 是指物理数据模型,与概念数据模型相比较,有一些基本概念已经发生了变化。如概念模型中的实体,被转化后将称为物理表;而部分关系根据关系数据库的转换原则,也将被转化为物理表;原有实体中的主属性将转化为物理表的主键,而相应的关系数据表中也将增加部分外键属性,从而使得关系数据表建立起父表与子表的逻辑关联。

第一步: 将概念模型 (CDM) 转化为物理模型 (PDM)

在 CDM 操作界面中,选择菜单 Tools 中的 Generate Physical Data Model (转换成物理模型),如图 7-21 所示。

在弹出的配置界面中配置导出的数据库管理系统 (DBMS) 为 Microsoft SQL Server 2005,并

设置 Name（显示名称）和 Code（物理数据库名称）为“成绩管理物理模型”，其余选项一律采用默认设置即可，单击“确定”按钮后开始转换。在 CDM 和 PDM 转换中，可能会出现错误，这些错误表现为一般错误和严重错误，一般错误并不影响 PDM 的生成，但是严重错误将无法转换为 PDM。在弹出的 Result List（结果列表）中，双击错误行，仍然进入 CDM 界面继续进行修改，直到 Result List 无报错为止，如图 7-22 所示。

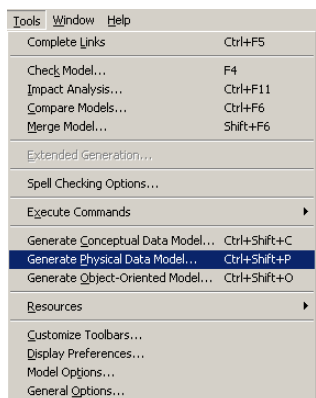


图 7-21 转换成物理模型



图 7-22 转换物理模型配置

第二步：配置物理模型（PDM）

按照概念模型转换物理模型的基本规律，原有的 CDM 中的实体转换成为物理表，对于多对多关系和部分一对一关系也直接转换成为物理表。对于一对多关系，一端实体的码加注到多端实体属性中成为外键。如学生和班级关系中，作为一端实体（班级表）的码（班级号码）会加入到多端实体（学生表）属性中成为外键。在显示中，主键表现为 PK（Primary Key），外键表现为 FK（Foreign Key）。考勤表是在多对多关系中直接生成的关系表。如图 7-23 所示为初次转化成为 PDM 图的基本对象。

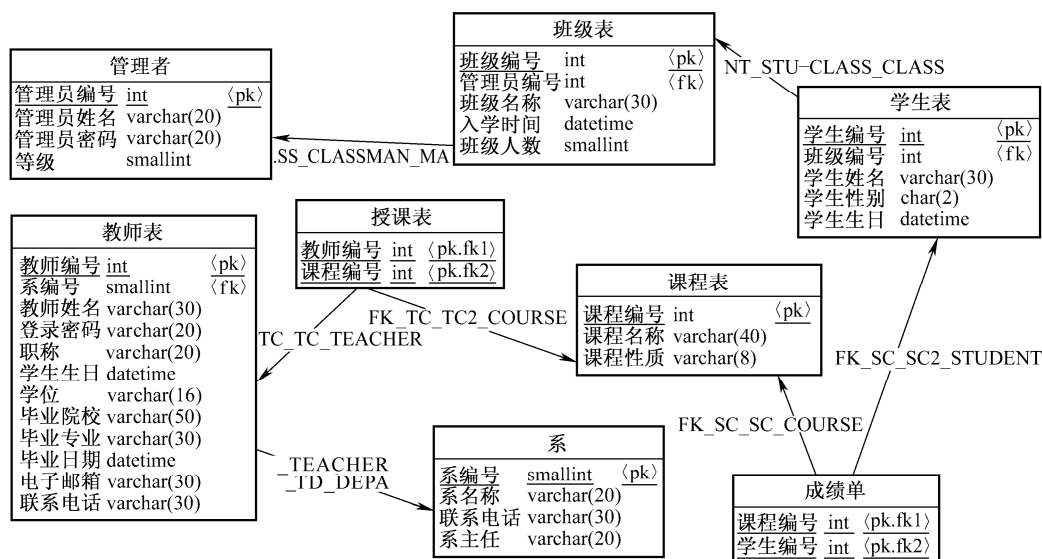


图 7-23 生成的物理数据模型图

生成的物理模型（PDM）是需要进行细致配置后才可以继续导入到数据库应用软件中的，配置的主要过程是配置物理表（table）。双击某个实体表，在弹出的配置界面中，首先设置物理表逻辑名称和编码名称，双击学生表对象，展开界面如图 7-24 所示。

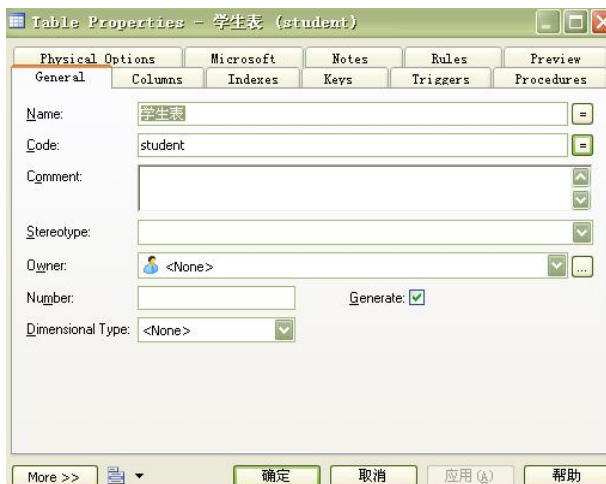


图 7-24 配置物理表界面

选择 Columns 选项卡，开始配置具体的表属性。同配置 CDM 图一样，Name 配置为中文名称，用以显示和说明备注；Code 为英文或拼音命名，将直接生成数据库的物理表名称，对于 Data Type（数据类型），可以选中其后的下拉菜单详细配置。对于有%n 情况的设置，可以手动键入具体的数值，如 varchar(%n)为变字长，可以配置成 varchar(20)，如图 7-25 所示。

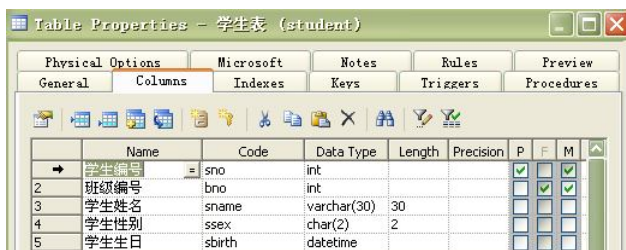


图 7-25 配置物理表属性界面

在设置界面中有 PFM 的配置选项，分别表示的意思是：P (Primary Key) 主键；F (Foreign Key) 外键；M (Mandatory) 必填项。当 P 被勾选后，则 M 同时被勾选，而 F 的设置是不可以由设置者配置的，它是通过配置表与表之间的关系时自动生成的。

对于多对多关系生成的关系表，由于是将两端表的主键合并成为关系表的码，因此这种类型的表的主键是由两个属性共同构成，同时它们也是外键，比如图 7-23 中的授课表和成绩单就是属于这种情况。当然联合属性作为主键并不是一种好的设计模式，一般我们最好再自定义一个属性作为主键，而去除原表中联合主键的特征。重新改造后的授课表和成绩单中，我们分别定义了新的主键，另外新增了一些非码属性，以便更加贴近实际。我们将这种机器自动生成的模式进行适当修改，既增加一个属性并使之变成主键，又不破坏原有的外键关系，目的是简化今后的 SQL 开发，如图 7-26 所示。

Figure 7-26 shows two screenshots of the 'Table Properties' dialog box in SQL Server Enterprise Manager. The top screenshot is for the '成绩单 (sc)' table, and the bottom screenshot is for the '授课表 (tc)' table. Both tables have columns defined with specific data types and constraints.

Name	Code	Data Type	Length	Precision	P	F	M
课程编号	cno	int			☒	☒	☒
学生编号	sno	int			☒	☒	☒
成绩单编号	scno	int			☒	☒	☒
考试成绩	grade	float			☒	☒	☒
考试属性	scprop	tinyint			☒	☒	☒
登记日期	dtime	datetime			☒	☒	☒

Name	Code	Data Type	Length	Precision	P	F	M
教师编号	tno	int			☒	☒	☒
课程编号	cno	int			☒	☒	☒
授课编号	tcno	int			☒	☒	☒
授课学期	skcq	varchar(30)	30		☒	☒	☒
课时数	sktime	tinyint			☒	☒	☒

图 7-26 改变属性后的授课表和成绩单

经过对每一张物理表和关系表的仔细设置和配置，得出系统最后的关系图（PDM），如图 7-27 所示，请同时将该 PDM 图与图 7-23 进行对比。

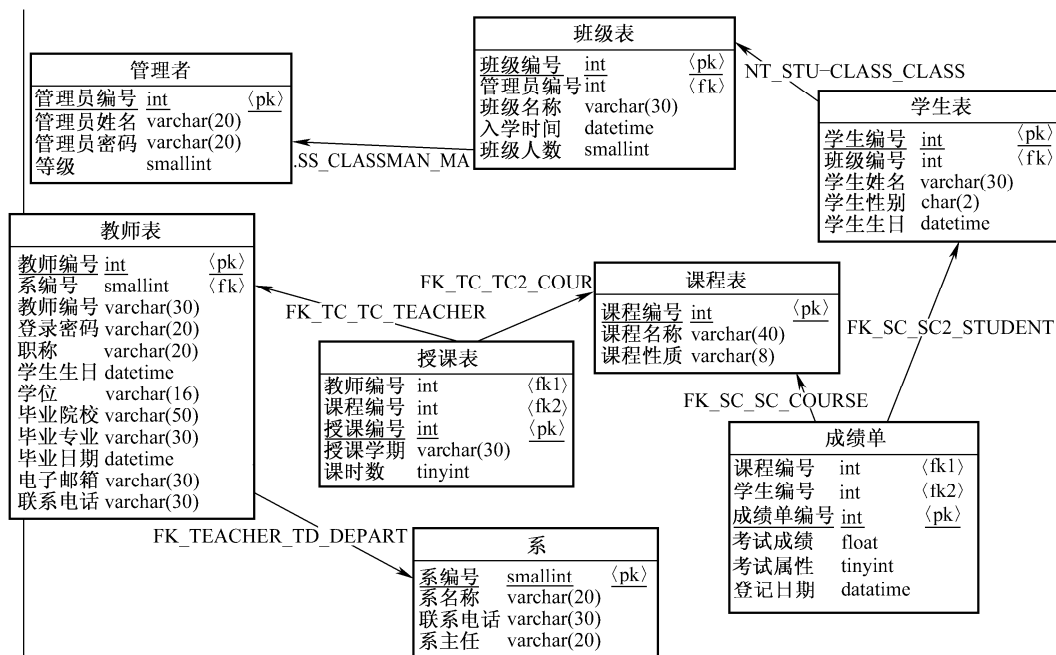


图 7-27 最后确定的 PDM

7-2-4 将物理模型导入到数据库应用软件中

经过以上 PDM 的细致设计工作，我们最后的目的是将设计的思想最后生成具体的数据库应用软件的物理表及其模式逻辑关系。下面的实例就是将上一节中的 PDM 图导入到 SQL Server 2005

之中，并完全实现设计的逻辑模式意图。



案例学习： 将教务成绩管理系统的物理数据模型（PDM）导入到 SQL Server 2005

第一步：配置生成实际数据库接口环境

选择菜单 Database 中的 Generate Database（生成数据库），如图 7-28 所示。

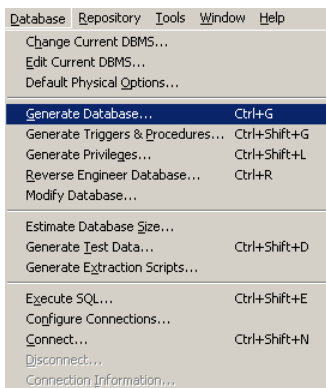


图 7-28 生成物理数据库

在弹出的配置界面中配置导出的数据库生成编码文件的路径（Directory）以及文件命名（File name），同时配置数据库的生成方式（Generation type）。Generation 的生成方式有两种：一种是 Script Generation（代码生成方式），另一种是 Direct generation（通过 ODBC 直接生成方式），如图 7-29 所示。



图 7-29 生成实际数据库配置界面

如果选择 Script Generation（代码生成方式），则将生成一个后缀名为.sql 的文本文件，该文件保存所有可在 SQL Server 的查询分析器下直接执行的代码，在 master 数据库环境下直接执行这些代码即可。如果选择 Direct generation（通过 ODBC 直接生成方式），则需要配置 Windows 操作系统的 ODBC 接口。

第二步：配置 Windows 操作系统的 ODBC 接口

首先在 SQL Server 2005 中建立一个新的空数据库，命名为 Grade。其次在 Windows 操作系统

的管理工具中设置 ODBC 数据源, 如图 7-30 所示。在“系统 DSN”选项卡中添加一个 SQL Server 数据源, 如图 7-31 所示。



图 7-30 设置 ODBC 数据源



图 7-31 添加一个 SQL Server 数据源

命名 ODBC 接口名称和数据库服务器, 在“服务器”栏目中键入(local)表示本机数据库服务器, 如图 7-32 所示。

选择运行在本机的刚刚新建的数据库 Grade, 如图 7-33 所示。



图 7-32 配置 SQL Server 数据源接口

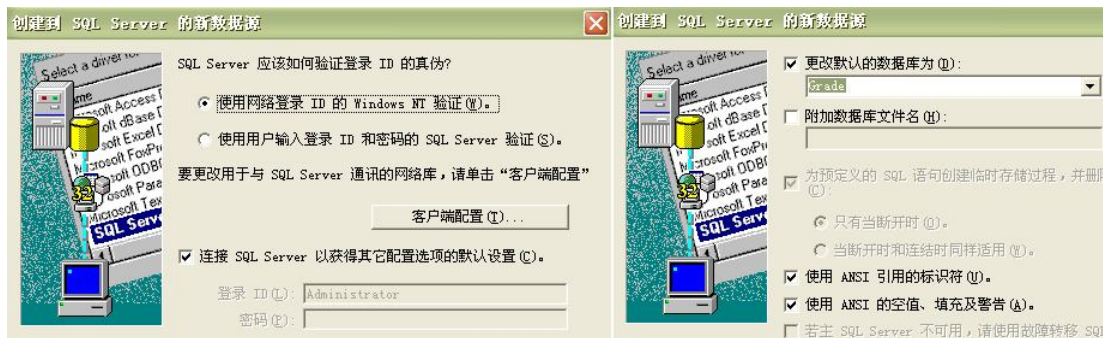


图 7-33 选择数据库 Grade 为指向的数据库

单击“下一步”按钮完成后，测试数据源是否成功，成功后即完成 ODBC 配置工作，如图 7-34 所示。

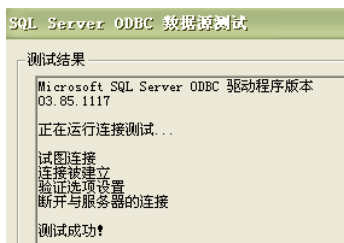


图 7-34 配置 ODBC 成功界面

第三步：将物理模型（PDM）生成实际数据库

回到 PowerDesigner 设计界面，选择 Direct generation，单击“确定”按钮后，经过系统数据库规范性校验后，弹出连接到数据源界面，选择刚才建立的 ODBC 命名的选项，如图 7-35 所示。

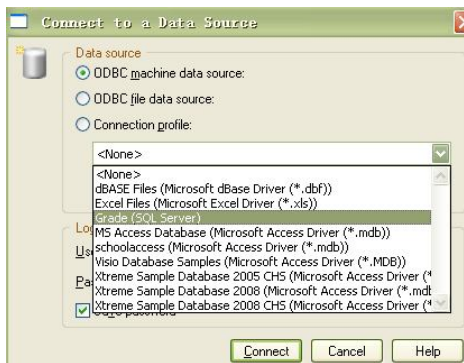


图 7-35 连接到数据源界面

单击 Connect(连接)后即可在 SQL Server 的数据库 Grade 中生成相关的表信息和约束及关系。需要注意的是，在生成期间会出现相关的运行问题，此时建议都单击 Ignore All（忽略全部），具体问题可以在 SQL Server 中继续进行修改，如图 7-36 所示。查看 SQL Server 的数据库 Grade，发现表已经全部生成。

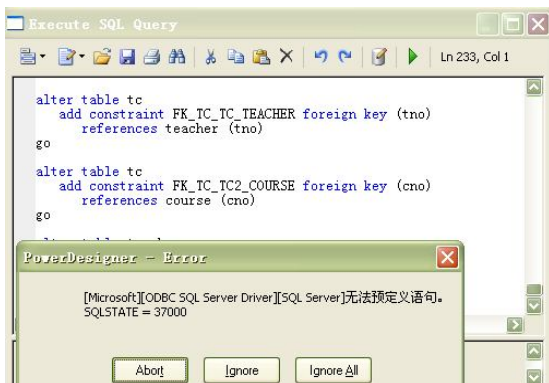


图 7-36 生成阶段出现问题时忽略

7-2-5 生成成绩管理系统数据库报告

很多时候,如果我们要将按照数据库建模思想生成的数据库告知项目中的每一个程序员,并细致地向他们说明复杂的逻辑关系,必须通过非常细致的数据库报告才可以办到,这一环节不仅费时而且费力。现在我们将学习通过 PowerDesigner 软件具有的强大数据库报告生成功能,完成以前这种麻烦的数据库报告工作。



案例学习：生成成绩管理系统数据库报告实例

1. 第一步：新建报告

在 PDM 设计管理界面中,选择菜单 Model 中的 Reports 选项,如图 7-37 所示。在报告列表中选择“New Report (新建报告)”,如图 7-38 所示。

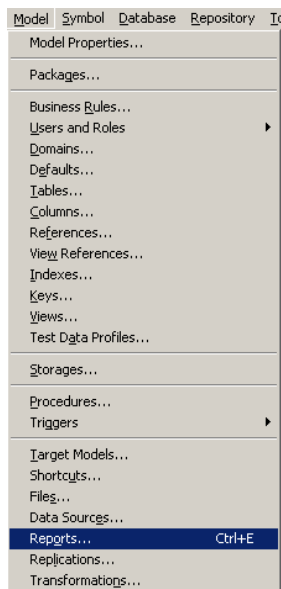


图 7-37 选择生成报告

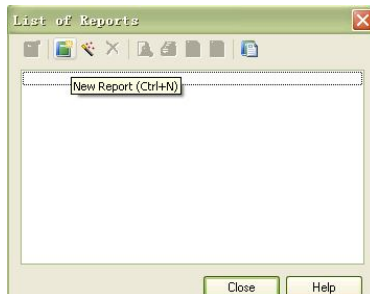


图 7-38 选择生成报告

2. 第二步：设置报告内容

在“新建报告”对话框中,键入报告名称为“成绩管理系统数据库报告”并选择具体的语言为“Simplified Chinese”,即简体中文,而后再选择报告生成的模板。模板分为 Full Physical Report (完全物理数据库报告), List Physical Report (物理数据库列表报告), Standard Physical Report (标准物理数据库报告)。此处选择 Full Physical Report,如图 7-39 所示。

设置完毕后将进入数据库报告设计界面,我们可以对封面、表头等信息进行具体设计,最后通过生成导航条选择生成的文件类型(包括 Word 格式的 RTF 文件或者网页格式文件 HTML),如图 7-40 所示。

选择后将自动生成相应格式的具体文件,用以进行数据库文档汇报,我们此次生成的是 HTML 数据库报告,如图 7-41 所示。



图 7-39 键入报告名和选择报告语言及模板



图 7-40 文档格式导航条



图 7-41 生成的 HTML 报告样式

7-3 PowerDesigner 与数据库建模实训



实训目标

- 通过 PowerDesigner 设计 CDM 图。
- 通过 PowerDesigner 设计 PDM 图。
- 将 PDM 结构导入到 SQL Server 2005 中，形成基本表和关系。
- 生成论坛系统数据库报告（Word 版和 HTML 版）。

数据库建模实训

1. 实训任务

某大型门户网站项目交付给你的任务是为一个论坛子系统进行数据库规划设计和建模的工作，具体数据表的实体关系图如图 7-42 所示。具体任务是：

- 1) 通过 PowerDesigner 设计出该论坛子系统的概念模型图（CDM）。
- 2) 设计出该论坛子系统的物理模型图（PDM）。
- 3) 将 PDM 结构导入到 SQL Server 2005 中，形成基本表和关系。
- 4) 生成论坛系统数据库报告（Word 版和 HTML 版）。

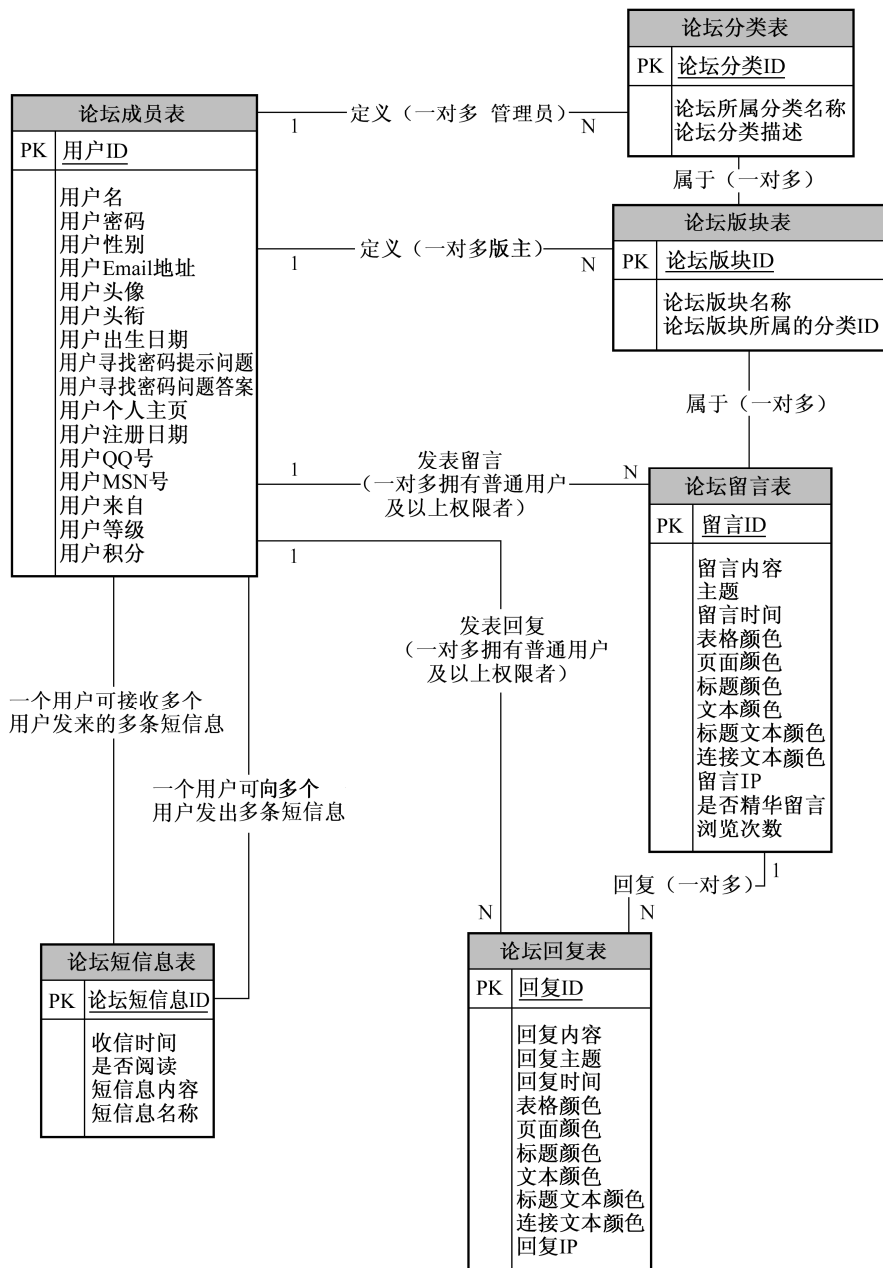


图 7-42 论坛数据库系统 BBS 实体关系图

2. 实训指导

在理解实体关系图的基础上,我们首先需要在 PowerDesigner 中建立 CDM 图,如图 7-43 所示。而后将该 CDM 图转化为 PDM 图,如图 7-44 所示。再将生成的 PDM 图导入到 SQL Server 2005 中,并生成报告,此处两步骤略,请读者参照本章相关内容部分自行完成。

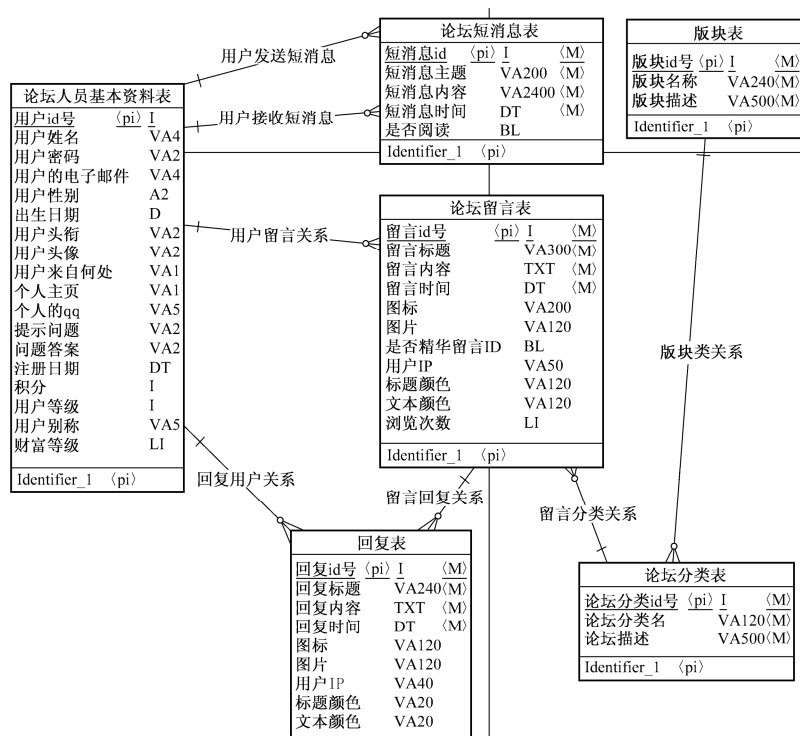


图 7-43 论坛数据库系统 CDM 图

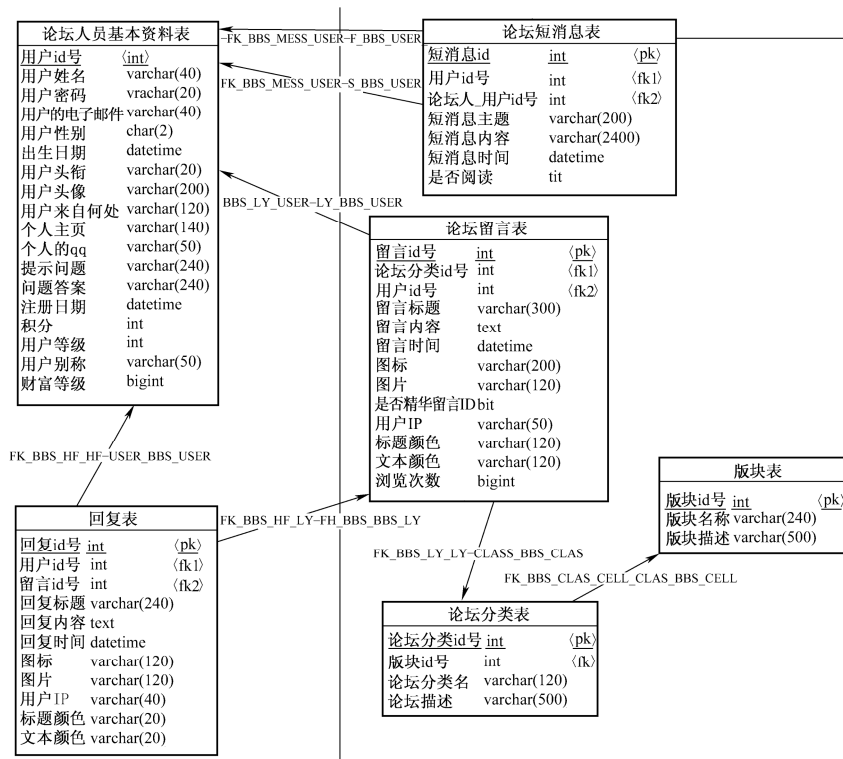


图 7-44 论坛数据库系统 PDM 图



本章考纲

- 了解数据库设计的要点。
- 掌握事实发现技术，基本步骤及完成数据库需求分析实例。
- 了解 PowerDesigner 基本的特性和发展历程，其主要功能和应用范围以及主要模块。
- 掌握通过 PowerDesigner 建立概念数据模型，认识其中的实体对象的属性、值域以及关键字，特别掌握实体之间关系的设计方法。
- 掌握通过 PowerDesigner 建立物理数据模型，掌握配置 PDM 图的全过程。
- 掌握将物理模型导入到数据库应用软件中的基本技术。
- 掌握生成数据库报告的基本技术。



课后练习

一、填空题

1. 数据库设计内容包括_____，_____和_____。
2. 所谓静态设计是指_____，根据给定应用环境，设计数据库的数据模型或数据库模式，包括_____和_____。
3. 动态特性设计是指确定数据库用户的行为和动作，即数据库的_____设计，包括设计数据库查询、事务处理和报表处理等。
4. 物理设计是指在选定的 DBMS 环境下，把静态特性设计中得到的数据库模式加以物理实现，即设计数据库的_____和_____。
5. PowerDesigner 主要包含 4 个模块，即_____、_____、_____和_____。
6. 在进行 CDM 设计时，每一行属性中都可以配置 MPD 选项。M (Mandatory) 表示_____；P (Primary Key) 表示_____；D (Display) 表示_____。
7. 在进行 CDM 设计时，对应法则关系一共有四种，分别表示的意思是：0,n _____，0,1 _____，1,1 _____，1,n _____。

二、简答题

1. 数据库设计为什么需要事实发现？
2. 事实发现技术的具体步骤包括哪些？
3. 描述 PowerDesigner 概念数据模型和物理数据模型的基本特点？彼此的区别是什么？
4. 简单描述将物理模型导入到 SQL Server 中的基本过程。