

第 3 章

应用实例技巧

本章讲述一些应用实例技巧,包括一对多表单设计、数据加锁方法、常用游标模板、通知发布和常用外部函数。表单设计是将数据关系转化成数据库中的表的过程。表单设计时,应该考虑到表间的操作约束、数据库的性能、数据的冗余控制等因素。数据加锁是一种保证事务正确执行的常用方法;游标模板可为编程人员在软件开发过程中提供借鉴和参考。通知发布有两重含义:一是在系统运行过程中,尤其在测试阶段,常常需要软件升级或数据整理,要求所有用户暂时停止使用系统,通知发布功能可以提示用户暂停使用系统或强行关闭系统运行,从而实现对系统的维护操作;其次通知发布还可以将企业的政策、新闻和会议通知等信息通过系统告知所有用户,这样发布信息,不仅节约信息传递成本,更重要的是提高信息传递的有效性和实时性。

教学要求

- (1) 了解表的关联关系;
- (2) 掌握一对多表单设计方法;
- (3) 了解数据加锁方法;
- (4) 掌握回滚与提示的语句顺序;
- (5) 理解游标模板含义并能应用游标模板;
- (6) 了解通知发布的作用和实现过程;
- (7) 了解常用外部函数。

重点

- (1) 一对多表单设计;
- (2) 数据加锁方法;
- (3) 回滚与提示语句顺序;
- (4) 游标使用。

难点

- (1) 通知发布的实现过程;
- (2) 常用外部函数。

3.1 一对多表单设计

3.1.1 关联关系

现实世界中相互可以区分的事物称为实体。实体之间存在着直接或间接的联系,这些联系在数据库中反映为实体之间的联系。两个实体之间的联系分为三类。

(1) 一对一联系(1:1)

对于实体集 A 和实体集 B,如果 A 中的每一个实体,在 B 中最多只有一个实体与之联系,反之亦然,则称实体集 A 与实体集 B 之间具有一对一联系,记作 1:1。例如,在大多数单位里,一个科室只有一个科长,而科长不允许兼职,则科长集与科室集之间具有一对一联系。

(2) 一对多联系(1:n)

对于实体集 A 和实体集 B,如果 A 中的每一个实体,在 B 中有 $n(n \geq 0)$ 个实体与之联系,反之,B 中的每一个实体,A 中有且只有一个实体与之联系,则称实体集 A 与实体集 B 之间具有一对多联系,记作 1:n。例如,一个科室有多个科员,而每个科员必须而且只能属于一个科室,则科室集与科员集之间具有一对多联系。

(3) 多对多联系($m:n$)

对于实体集 A 和实体集 B,如果 A 中的每一个实体,在 B 中有 $n(n \geq 0)$ 个实体与之联系,反之,B 中的每一个实体,在 A 中也有 $m(m \geq 0)$ 个实体与之联系,则称实体集 A 与实体集 B 之间具有多对多联系,记作 $m:n$ 。例如,一个学生可以选修多门课程,而任何一门课程也可以被多个学生选修,则学生集与课程集之间具有多对多联系。

3.1.2 数据设计模型

数据模型是现实世界中数据特征的抽象。数据模型应该满足三个方面的要求:一是能够比较真实地模拟现实世界,二是容易为人所理解,三是便于计算机实现。

1. 关联符号

在关系型数据库中,现实世界中的实体与实体间的对应关系被抽象为表单之间的关联关系。因此,表单之间存在一对一(1:1)、一对多(1:n)和多对多($m:n$)的关联关系。

在关系型数据库概念模型中,常用的关联关系符号如图 3.1 所示。图(a)表示可选,(b)表示强制,(c)表示一对多,(d)表示依赖。依赖是一对多关系的一种,要求左侧表的关键字必须作为右侧表的主键属性。

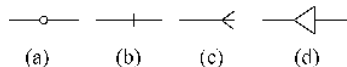


图 3.1 关联关系符号

2. 概念数据模型

概念数据模型(Conceptual Data Model, CDM)以实体-联系(Entity-Relationship, E-R)理论为基础,并对这一理论进行了扩充。它从用户的角度出发对信息进行建模,主要用于数据库的概念级设计。

CDM 是一组严格定义的模型元素的集合,这些模型元素精确地描述了系统的静态特性、动态特性和完整性约束条件,它包括数据结构、数据操作和完整性约束三部分。数据结构表达为实体和属性;数据操作表达为实体中记录的插入、删除、修改、查询等操作;完整性约束表达为数据的自身完整性约束(如数据类型、检查、规则等)和数据间的参照完整性约束(如联系、继承联系等)。

3. 物理数据模型

物理数据模型(Physical Data Model,PDM)是在概念数据模型的基础上,考虑各种具体实现因素,进行数据库体系结构设计,真正实现数据在数据库中的存放。建立物理数据模型的主要目的是将 CDM 转化成特定数据库管理系统下的 SQL 程序,PDM 是 SQL 程序的图形化表示。PDM 能够在特定的数据库管理系统中建立用于存放信息的数据结构(如表、约束等)。

部门与员工之间的关联是一个典型的一对多关联关系实例。“部门档案”表为父表,其主键为“科室名称”(带下划线的属性为主键,下同)，“员工档案”表为子表,其主键为“科室名称”+“员工姓名”,其概念模型如图 3.2(a)所示。由于定义了由子表到父表的强制依赖关系,因此“科室名称”会自动被定义为子表主键的一部分。当向“部门档案”表中添加新科室时,“员工档案”表中一定还不存在该科室的员工;但向“员工档案”表中添加新员工工时,“部门档案”表中一定已经存在该员工所在的科室。图 3.2(b)是其对应的数据库物理模型。

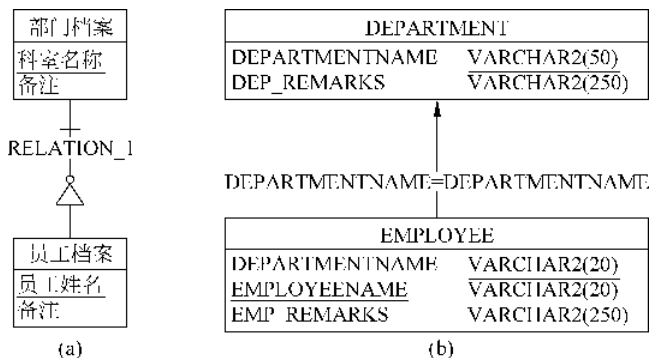


图 3.2 一对多关联关系

4. 面向对象模型

面向对象模型(Object Oriented Model,OOM)是运用面向对象环境将模型信息用标准化的图形元素进行显示并建立软件模型的过程。它在不同的层次上显示系统的工作状况,是表达软件系统含义的强有力的工具。它便于分析人员、用户、开发人员、测试人员和管理人员之间交流信息。

面向对象建模的一个重要问题是采用哪种图形标注方法来表示系统的各个方面。目前常用的图形标注方法有 Booch 方法、对象建模技术(OMT)和统一建模语言(UML),其中 UML 是大多数公司采用的标准。UML 中包含用例图、时序图、类图、协作图、对象图、状态图、活动图、组建图、部署图等 9 种。

进行数据库设计主要用到前两种模型。概念数据模型是数据建模的第一个阶段,它把

现实世界中的信息抽象成实体和联系来产生实体联系图,这一阶段为高质量的应用提供坚实的数据模型基础。概念数据模型与具体的数据库系统、操作系统平台等无关。物理数据模型主要解决现实世界中信息在数据库管理系统中的存储结构问题,同一个概念数据模型结合不同的数据库管理系统将产生不同的物理数据模型。物理数据模型是后台数据库应用的蓝本,直接针对具体的数据库管理系统。

3.1.3 一对多表单数据库设计

数据库设计是信息系统建设过程的一个重要环节,它是根据企业的业务需求、信息需求和处理需求来确定信息系统的数据库结构、数据操作和数据一致性、完整性约束的过程。

对于大多数信息系统开发人员来说,能够从用户那里得到的需求信息往往是零乱并且错综复杂的,这些信息大多以用户企业往来业务中的单据或手工报表和记录单来体现。用户以规范化数据库结构来描述这些信息是不大可能的,开发人员更是不能奢望用户根据这些零乱并且错综复杂的单据或手工报表和记录单提出未来信息系统的实现过程和操作模式。因此,以下三个问题是所有系统开发人员必须面对而且必须得到妥善解决的问题:

(1) 从用户提供的零乱并且错综复杂的需求信息中抽象并优化,设计出符合规范化要求的数据库结构。

(2) 在设计信息系统的数据库概念模型和物理模型中,要考虑用户过去手工处理与信息系统新的数据存储结构的差异和系统实现等多方面的因素。

(3) 设计并实现尽量符合用户手工操作习惯的信息系统操作功能界面。

下面以医院信息管理系统中药库出入库业务的设计与实现为例说明一对多表单设计与优化的过程,其原始单据见图 3.3。图中用“*”表示的数据为实际数据,但此处未给出实际值。

药库入库单

入库单编号: 2682

入库日期: 2007.11.30

入库人员: 李 平

供应商名称: 神州制药公司

地 址: 北京

邮 编: 100 ***

供应商开户行: 工商银行

供应商银行账号: 0 * *-8 *****

供应商电话: 010-8 *****

联 系 人: 李 东

联系电话: 133 ***** 0981

序号	药品编码	药品名称	单位	包装	进价	数量	金额	有效日期
1	05043	10%葡萄糖注射液(250ml)	瓶	箱	6.00	1000	6 000.00	2008.02.30
2	01041	阿莫西林胶囊(250mg×24 粒)	盒	盒	12.00	1 000	12 000.00	2008.02.30
3	01076	阿昔洛韦片(0.1g×30 #)	盒	盒	22.00	100	2 200.00	2008.04.30
4	01054	阿奇霉素片(250mg×6 粒)	盒	盒	30.00	50	1 500.00	2008.04.30
5	03014	安宫黄体酮片(2mg×100 #)	盒	盒	15.00	200	3 000.00	2009.06.30
6	09020	爱活胆通胶囊(100 粒)	盒	盒	20.00	260	5 200.00	2009.06.30
7	05044	5%葡萄糖注射液(250ml)	瓶	箱	5.00	1 000	5 000.00	2009.08.30
8	02023	安体舒通片(20mg×100 #)	盒	盒	120.00	50	6 000.00	2010.08.30
9	13034	84 消毒液(500ml)	瓶	箱	8.00	100	800.00	2010.10.30
10	07010	安定片(2.5mg×100 #)	瓶	瓶	150.00	20	3 000.00	2010.10.30
总金额: 44 700.00								
审核人: 王 中			审核日期: 2007.12.01					

图 3.3 药库入库单

1. 关系的抽象和优化

关系的抽象是指把用户提供的单据、手工报表和记录单等能够反映企业实际往来业务的数据转化为数据库表的过程；而关系的优化是指对抽象以后的数据库表进行规范化处理的过程，以使其能够满足最小数据冗余度、数据相对独立性、数据完整性和数据一致性等数据库设计的规范化要求。

关系型数据库对数据结构有第一范式(1NF)、第二范式(2NF)和第三范式(3NF)等规范化定义。为了表示其规范化的程度，可以将其称为一级规范、二级规范和三级规范，其规范化程度依次增强。关系的优化过程通常是通过对表的拆分和合并依次实现从一级规范到二级规范、从二级规范再到三级规范的转化。

一级规范要求二维表中不能存在组项(可分割的非原子项)、空白项和重复项。因此，对于系统开发人员来说，第一项工作就是将图 3.3 所示的反映药品入库业务的药库入库单抽象为二维表结构，如表 3.1 所示。表 3.1 的所有列都是不可再进行拆分的原子项，且不含空白项和重复项，所以可以将其抽象为数据库表，如图 3.4 所示，它是符合一级规范要求的，但是它存在着非常严重的数据冗余。

二级规范要求在一级规范化后的表中不能存在部分依赖关系的属性(非主键属性依赖于主键的一部分)。在图 3.4“入库单”表中，主键由“入库单编号”和“序号”两个属性组成，而从“药品编号”到“有效日期”8 个属性都只依赖于“序号”属性(主键的一部分)，因此需要将这些属性从“入库单”表中拆分出来，形成“入库单单头”和“入库单单目”两个表，如图 3.5 所示。“入库单单头”表和“入库单单目”表之间是依赖关系，在一个“入库单单头”中

入库单
<u>入库单编号</u>
<u>序号</u>
入库日期
入库人员
供应商名称
联系人
联系电话
供应商电话
地址
邮编
供应商开户行
供应商银行账号
药品编码
药品名称
单位
包装
进价
数量
金额
有效日期
总金额
审核人
审核日期

图 3.4 一级规范后的“入库单”结构

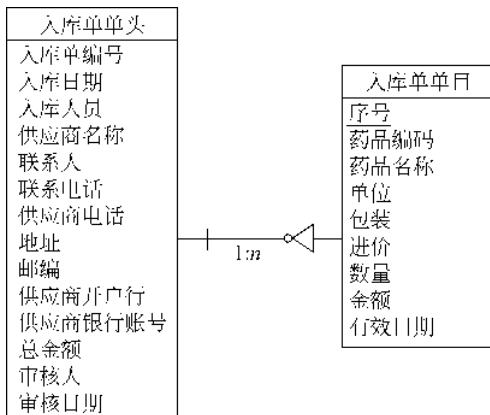


图 3.5 二级规范后的“入库单”结构

表 3.1 一级规范的药库入库单表

入库单 编号	入库 日期	入库 人员	审核 人	审核 日期	供应商 名称	地址	邮编	开户 行	银行 账号	电话	联系 人	联系 电话	序 号	药品 编码	药品 名称	单 位	包 装	有效 日期	进 价	数 量	金 额
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	1	05043	10%葡萄糖注 射液(250ml)	瓶	箱	2008. 02.30	6.00	1 000	6 000.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	2	01041	阿莫西林胶囊 (250mg×24粒)	盒	盒	2008. 02.30	12.00	1 000	12 000.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	3	01076	阿昔洛韦片 (0.1g×30#)	盒	盒	2008. 04.30	22.00	100	2 200.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	4	01054	阿奇霉素片 (250mg×6粒)	盒	盒	2008. 04.30	30.00	50	1 500.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	5	03014	安宫黄体酮片 (2mg×100#)	盒	盒	2009. 06.30	15.00	200	3 000.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	6	09020	爱活胆通胶 囊(100粒)	盒	盒	2009. 06.30	20.00	260	5 200.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	7	05044	5%葡萄糖注 射液(250ml)	瓶	箱	2009. 08.30	5.00	1 000	5 000.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	8	02023	安体舒通片 (20mg×100#)	盒	盒	2010. 08.30	120.00	50	6 000.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	9	13034	84 消毒液 (500ml)	瓶	箱	2010. 10.30	8.00	100	800.00
2682	2007. 11.30	李平	王中	2007. 12.01	神州制 药公司	北京	100 ***	工商 银行	0 *** 8 *****	010- 8 *****	李东	133 ***** 0981	10	07010	安定片 (2.5mg×100#)	瓶	瓶	2010. 10.30	150.00	20	3 000.00

可以有多个“入库单单目”(每个单目表示一种药品,药品编码可以重复,药品名称可以重名),二者之间是一对多(1:n)的关联关系。经过这一步的优化后,表 3.1 中的数据存储可以转化为两个二维表的数据存储,分别见表 3.2 和表 3.3。两个表格之间通过“入库单编号”实现一对多(1:n)的关联关系。针对图 3.3 这张药库入库单,比较表 3.1 和表 3.2、表 3.3 的数据存储结构和规模,可以看出,表 3.2 和表 3.3 的数据存储量要比表 3.1 小很多,主要是减少了对入库单单头信息和单脚信息的冗余存储。

从以上对数据结构的二级规范化处理过程可以看出,二级规范化的主要目的是在最大程度上降低数据的冗余度和数据一致性维护的代价。

表 3.2 药库入库单单头表

入库单编号	入库日期	入库人员	审核人	审核日期	供应商名称	地址	邮编	开户行	银行账号	电话	联系人	联系电话	总金额
2682	2007.11.30	李平	王中	2007.12.01	神州制药公司	北京	100 ***	工商银行	0 ** - 8 *****	010- 8 *****	李东	133 **** 0981	44 700.00

表 3.3 药库入库单单目表

序号	入库单编号	药品编码	药品名称	单位	包装	有效日期	进价	数量	金额
1	2682	05043	10%葡萄糖注射液(250ml)	瓶	箱	2008.02.30	6.00	1 000	6 000.00
2	2682	01041	阿莫西林胶囊(250mg×24 粒)	盒	盒	2008.02.30	12.00	1 000	12 000.00
3	2682	01076	阿昔洛韦片(0.1g×30 片)	盒	盒	2008.04.30	22.00	100	2 200.00
4	2682	01054	阿奇霉素片(250mg×6 粒)	盒	盒	2008.04.30	30.00	50	1 500.00
5	2682	03014	安宫黄体酮片(2mg×100 片)	盒	盒	2009.06.30	15.00	200	3 000.00
6	2682	09020	爱活胆通胶囊(100 粒)	盒	盒	2009.06.30	20.00	260	5 200.00
7	2682	05044	5%葡萄糖注射液(250ml)	瓶	箱	2009.08.30	5.00	1 000	5 000.00
8	2682	02023	安体舒通片(20mg×100 片)	盒	盒	2010.08.30	120.00	50	6 000.00
9	2682	13034	84 消毒液(500ml)	瓶	箱	2010.10.30	8.00	100	800.00
10	2682	07010	安定片(2.5mg×100 片)	瓶	瓶	2010.10.30	150.00	20	3 000.00

三级规范要求二级规范化后的表中不能存在传递依赖关系的属性(非主键属性依赖于另一个非主键属性,而此属性依赖于主键)。在图 3.5 满足二级规范的“入库单”结构中,“入库单单头”表的主键是“入库单编号”,“供应商名称”依赖于主键“入库单编号”,而“联系人”、“联系电话”、“供应商电话”、“地址”、“邮编”、“供应商开户行”、“供应商银行账号”7 个属性依赖于“供应商名称”,并不直接依赖于主键“入库单编号”,这是典型的传递依赖关系。要使其满足三级规范的要求,就必须把这 7 个属性从“入库单单头”表中拆分出来组成一个新的“供应商档案”表,二者之间通过“供应商名称”属性实现一对多(1:n)关联。

类似地,在“入库单单目”表中,“药品编码”依赖于“入库单编号”+“序号”,而“药品名称”、“单位”、“包装”3 个属性仅仅依赖于“药品编码”,也存在传递依赖关系,就必须把这 3 个属性从“入库单单目”表中拆分出来组成一个新的“药品资料档案”表。所有涉及药品资料数据处都会用到该数据库表,如开处方和药房管理业务等。

再仔细分析图 3.5“入库单单目”表,“金额”属性依赖于“进价”和“数量”,并不直接依赖于主键,也属于需要消除的传递依赖。实际上,“金额”属性是由“进价”和“数量”相乘得到的

导出属性,可以将其删除,这样做的好处表现在既可以降低数据冗余,又可以有效地降低修改异常,增强数据的一致性。进一步观察图 3.5 中的“入库单”结构,可以看出“入库单单头”表中的“总金额”属性实际上是经过“入库单单目”表中所有单目的“数量”×“进价”再进行合计派生出来的属性,该属性的存在理论上不但增加了数据冗余度,而且极易造成数据的不一致,因此将其从“入库单单头”表中剔除。图 3.6 是经过三级规范化后的“入库单”结构。

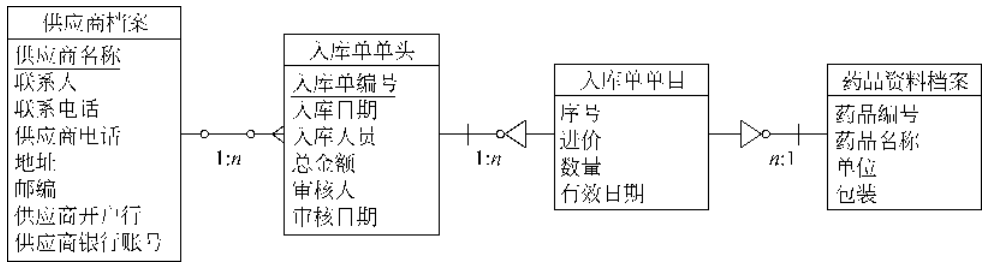


图 3.6 三级规范化后的“入库单”结构

2. 数据库模型

经过抽象和优化后的数据库结构从理论上来说,已经满足了数据库概念模型和物理模型的设计与实现要求,但是它同实际系统开发所需要的数据库结构相比较,还有一定的距离。

从用户的使用角度出发,以信息系统来实现企业相关业务的管理,其最大的优点之一,表现在其强大的信息存储能力和信息的检索、统计以及分析等处理能力。因此,用户往往希望所开发的信息系统能够存储比手工业务处理过程更加全面的信息,拥有比手工业务处理过程更加强大的信息检索、统计以及分析等处理能力。在医院药库出入库业务的实现中,如图 3.6 所示的优化后的“药库入库单”结构,体现出来的手工处理过程中的供应商信息包括“供应商名称”、“联系人”、“联系电话”、“供应商电话”、“地址”、“邮编”、“供应商开户行”、“供应商银行账号”等。但在欲开发的信息系统中,加入能够全面记录供应商的“法人代表”、“总经理”、“注册资金”、“注册日期”、“企业类型”、“经营方式”、“经营范围”、“上级主管”、“发证机关”、“传真”、“Email”、“HomePage”等信息。对于药品资料档案信息,除了入库单中的“药品编码”、“药品名称”、“单位”和“包装”外,加入能够全面记录药品资料档案的“规格”、“批次号”、“出厂日期”、“分类”、“甲乙类”、“药品级别”等信息。

从系统开发人员进行系统实现的角度出发,需要设置一些能够记录系统操作痕迹和一些有利于系统实现的属性。在此例中,为了对任何操作人员在操作系统的过程中记录操作员的操作信息,需要对所有表设置“记录日期”和“操作员”属性;为了记录表中记录的不同状态(如出入库单的执行状态有“未审”和“已审”等)和可能出现的随机补充备注信息,需要对一些表设置“状态”、“标记”和“备注”属性;为了适应药品在医院信息管理系统中以批发单位入库,以零售单位出库的业务需求,需要在“药品资料档案”表中设置“批发单位”、“零售单位”和二者之间转化的“换算量”属性,同时剔除原有的“单位”属性;为了提高用户在录入时的操作效率,需要在药品资料档案中设置“拼音码”属性以提高对药品资料的检索速度;

为了在一张表能同时实现药品的出入库业务,需要在“出入库单单头”表中设置“出入类型”属性,同时将原有的“供应商名称”属性更改为“供应接收单位”属性;另外,考虑到对出入库单信息进行统计汇总、查询、分析时,往往只需要单头信息,为了提高系统的查询效率,需要把结构优化处理时所剔除的“总金额”属性再添加到“出入库单单头”表中。

因此,对能够反映企业实际业务的单据或手工报表和记录单,经过抽象和优化后所得到的数据库结构,尽管它已经满足了三级规范的要求,但是在进行正式数据库结构设计时,还要充分考虑用户对欲建立的信息系统的数据存储期望和有利于系统实现的实际需求。有时为了增强系统的可实施性,尤其是提高系统的执行效率,经常需要增加一些冗余属性而与数据结构的规范化要求相悖。这些情况在这个实际案例中都有一定程度的体现。如从图 3.4~图 3.6 可以看出,使用二级规范的结构相对于一级规范结构大幅度减少了数据冗余,同样使用三级规范相对于二级规范也减少了数据冗余。经过以上设计处理后,最终确定的医院出入库业务的数据库概念模型如图 3.7 所示,其物理模型如图 3.8 所示。

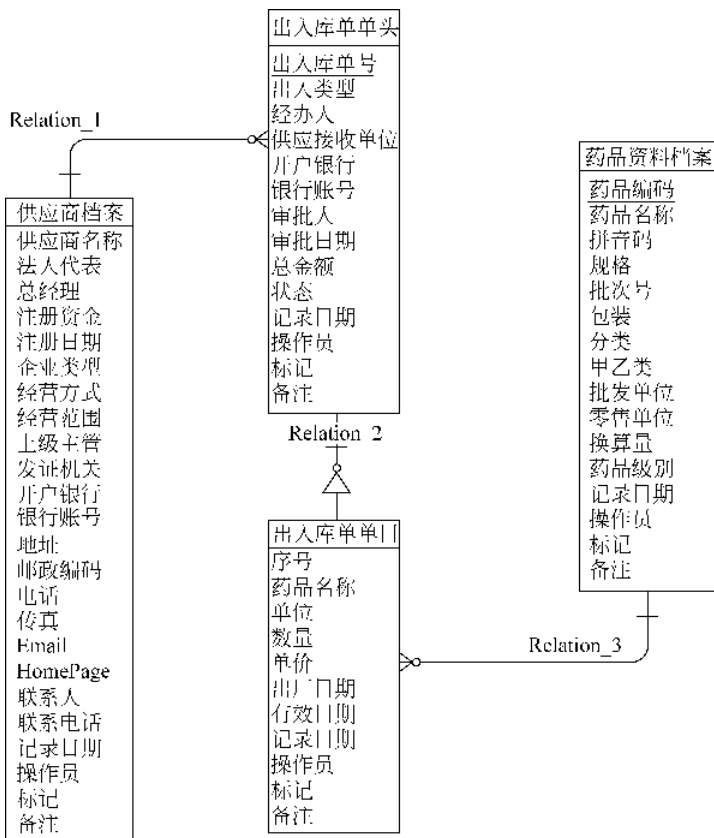


图 3.7 药品出入库业务数据库概念模型

在图 3.7 中,“供应商档案”和“药品资料档案”属于档案信息表,“出入库单单头”和“出入库单单目”属于账务信息表。从业务角度讲,账务信息表需要长期保存且在审核后任何数据不允许被修改,即使数据有误也只能通过再补一张账表的方法去弥补以前的错误。从技术角度讲,当修改档案信息表时,根据一致性原则会自动修改账务信息表,这样很可能形成长事务,且有些账务信息表的数据已转入历史数据库无法修改。

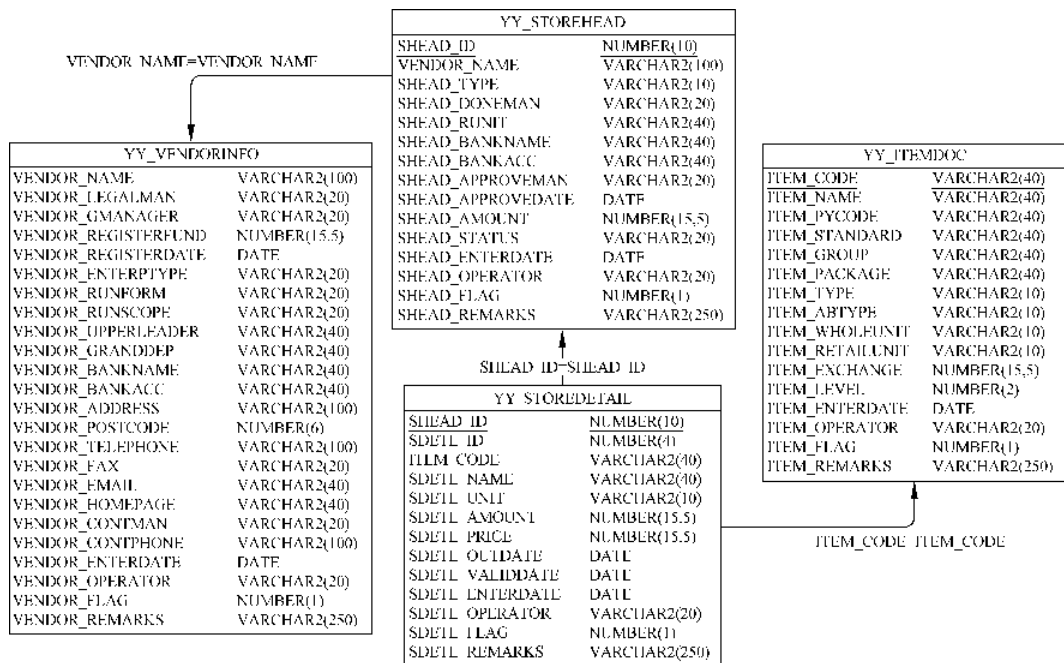


图 3.8 药品出入库业务数据库物理模型

综上所述,通过修改档案信息表来级联修改账务信息表的做法是不现实的,尤其是对一些长时间使用过的账务表数据修改而引发的长事务问题更为突出。综合考虑的解决方案如下:

(1) 每次出入库审核后,打印保存纸质账单,并以此记录为准,但这种做法与信息系统的目标有一定的冲突。

(2) 对档案表设有审核过程,一旦审核通过将不允许修改,且已用过的主键数据不能重复使用,如“药品资料档案”表中的“药品编码”一旦用过,则再也不使用,也不再将其分配给一种其他新药使用。

(3) 出入库审核后打印并保存纸质账单,使用二级规范表结构,档案信息表仍然存在,但为了数据的一致性,录入数据时,需要用档案信息表对应的数据填充账务信息表的相应数据。当“出入库单单头”和“出入库单单目”信息已通过审核后,Relation_1 和 Relation_3 无级联修改和删除关系,即对“供应商档案”信息的修改或删除不影响“出入库单单头”表中的相关数据,“药品资料档案”的修改或删除不影响“出入库单单目”表中的相关数据。当“出入库单单头”和“出入库单单目”信息未通过审核时,Relation_1 和 Relation_3 存在级联修改和删除关系。

主键是能够表示记录唯一性的属性集合。选择主键属性时尽量使用表中能反映记录特征的属性,且使查询尽量方便,否则可为此设置专门的属性,如图 3.7 的“出入库单单目”表中“序号”属性就是为此目的设置的。另外,在选择主键属性时要考虑属性的数据类型,如 blob 类型不能作为主键属性,逻辑型由于取值只有两个值,因而很少作为主键属性,实数由于其计算机表示的特殊性,最好不要作为主键属性,否则很可能造成使用等于条件无法查询到目标记录的现象。

由于表属性个数受数据库管理系统限制,如 Oracle 不能超过 256 个,所以对于数据项个数过多,尤其是数据项个数不定的关系应将数据分类。一类是关系的特征数据项;另一类是非特征数据项。对于特征数据项,每个数据项可作为数据库表的属性;而对非特征数据项可以按其反映的业务类型将其归并成几个可变长字符串或一个 blob 类型。如试油业务数据由反映油井的地区、井号、类型、层序、数据点数等特征数据项和数据点数对应的(曲线坐标)非特征数据项构成。地区、井号、类型、层序、数据点数等数据项可作为数据库表的属性,而曲线坐标数据项少则几十项,多则几百项,一个坐标数据由两个实数表示,占 8 字节,一条记录的全部坐标数据归并后不超过 10KB,可用 1~3 个变长字符串或一个 blob 类型表示。

图 3.9 是一个试油数据库系统对数据进行分类后归并的例子。图中位于下面的窗口显示的数据属于特征数据,上部窗口显示的曲线是用下面窗口对应记录的非特征数据展开后绘制的。一般地,用户首先浏览特征数据,对关心的油井才进行绘图操作,因为绘图操作用时较长。

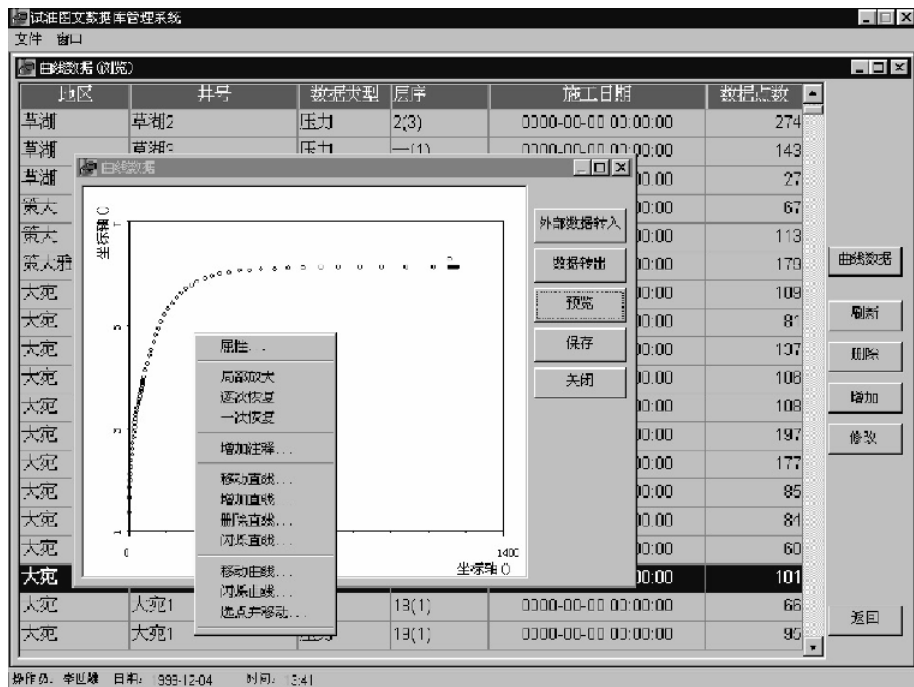


图 3.9 数据分类显示

当表自身和自己存在一对多关系时,其一致性维护不能靠触发器机制实现。

对于存在一对一关系的两个表可以合成一个表,但是当其中的一个表出现大量的空数据时,使用两个表能节约存储空间。对于存在多对多关系的两个表,可以通过引进一个中间表将其分解成两个一对多关系表。如“课程”与“学生”之间存在多对多关系,一门课程可以由多个学生选,同时一个学生可以选修多门课程,其关系如图 3.10(a)所示。通过引进中间表“学生选课”表,原多对多关系变成了两个一对多关系,如图 3.10(b)所示。图 3.10(a)和图 3.10(b)的关系是等价的,对应的物理模型见图 3.10(c)。