关系数据库 逆向工程 软件再工程 面向对象

计算机科学 2000Vol. 27№ 12

基于面向对象模型的关系数据库逆向工程研究

Relational Database Reverse Engineering Based on Object-Oriented Model

114-117

马 恕

（中央财经大学信息管理系 北京 100081）

余永红 徐洁磐

（南京大学计算机科学与技术系 南京 210093）

TP311.132 TP311.5

Abstract Recently, researchers have paid more attention on software reverse engineering, this paper mainly discusses the basic problems of relational database reverse engineering based on object-oriented model, it also presents a method which can be used to implement relational database reverse engineering based on EXPRESS object-oriented model. The research can play an important role on improving the quality of software reengineering, and extending the domain of object-oriented methodology.

Keywords Relational databse, Reverse engineering, Object-oriented model, EXPRESS

1 引言

逆向工程在软件再工程，在支持向客户/服务器体系结构的迁移，在了解分布式系统的状态和演化等许多方面正日益被认为是一种有效而经济的实现方式，逆向工程在处理软件和遗传系统方面的能力，也正得到软件组织越来越多的认可和赞许。逆向工程的核心是抽取抽象的活动，其中对数据的抽取处理是逆向工程的主要内容。由于在许多面向数据的应用系统中，其基础部件是数据库且数据库是系统中相对较为稳定的部分，因此对数据库结构的抽取是重要的一个方面，它的抽取可有利于对其它过程抽取的进行。

目前已有许多关系数据库逆向工程的论述[1~6]，但这些文章大都基于实体-联系模型而很少有基于面向对象模型的研究，由于面向对象模型可以提供方便进行软件再工程的自然描述，一个面向对象模型可以描述已有软件，实现逆向设计的语义目的，并正向设计一个新系统，面向对象模型非常适合于关系模式的逆向分析的表达，因此本文只讨论基于面向对象模型的关系数据库逆向工程的问题。

2 面向对象模型与关系数据库逆向工程

面向对象模型的基本概念是对象，我们把具有相同属性和方法的对象集合在一起便称为类，类与类之间存在继承和合成关系并构成相应的类层次结构，面向对象模型涉及继承、封装和抽象等面向对象概念，具有较强的灵活性、可扩充性和可重用性。而关系模型的概念比较单一，其概念模型是 E-R 模型或称为实体-联系模型，它将现实世界的要求转化成实体、属性、联系等几个基本概念及它们之间的几种基本关系，实体以及实体之间的关系通过关系来表达。

两种模型概念及表示上的不同导致从一个模型向另一个模型转换的困难，同时由于在基于关系数据库的开发过程中形成的关系数据库模式可能根据实际需要进行了优化、或者是数据库模式设计者的无意疏忽或设计人员的表述存在问题、因此最后建立的物理关系数据库模式很可能与设计时不一致，从而导致逆向工程的复杂化，一般基于面向对象模型的关系数据库逆向工程需要解决的问题包括：

(1)表：典型地每张表映射成一个类、但有时会出现一个类对应多张表或多个类对应一张表的情况。一个类可以从水平、垂直或两者兼而有之的方式分解成多张表。

(2)主键及主键标识：一般每张表都有一个主键，但在主键的标识方法上仍存在不同的实现选择，许多关系数据库系统除属性值外，还通过提供函数帮助来实现对象标识；而面向对象模型是通过对象标识符来标识对象的。

(3)关系：多对多的关系通常用一个特定的联系表来实现，一对多或一对一的关系也可用一个关系表或作为一个外关键字来实现，对一对多关系来说，外关键字放在多类中，对一对一的关系来说，外关键字可放在任意一个类中。有关联系的种类包括：①双向关系联系：在许多情况下，我们发现关联被放在参与关联的两个类中，这是一个合理的设计技术，因为关联可以在双向作快速查询，但这种构造方式使逆向设计复杂化了，猛一看双向关联象两个独立的关联，进行数据分析时可以测出双向指针的冗余，但对语义的理解需要解决这种情况。②可选型关联：在多对多或一对多的关系中，由于在某方可能出现，也可能不出现属性值，而实

际上这两种情况都应产生严格一致的模式。③可替换型关联：一个表从另一个表中和一个规格说明中推导出标识，开始时我们把这种情况考虑为两个分离的关联。④三重和多重关联：人们在构造应用模型中很少使用三重或多重关联，但实际中当人们进行数据库逆向工程时却经常发现三重关联，通过从三个或更多类中的主键进行合并标识一个关联表中的记录。若基于面向对象模型进行数据库逆向工程，需要解决关系模型中关系的表示问题。

(4)模式分解：逻辑结构有时对违背范式的形式合并，有时违背范式是可接受的，但作出这种决定应有一定的原因。模式分解的方式有：①多类表：一张表可能以交互关联的方式合并两个或多个实体，这样一张表可以也可以不满足范式要求。②继承的分解：目前关系数据库缺乏对继承的语义支持，一种策略是定义单独的超实体和子实体表，这种策略的引用完整性可以强制子实体和超实体之间的相应关系，但关系数据库不能表达这种普化的分解，即超实体中的每个对象在某一子实体中被进一步描述，人们可把属性向下传递给子实体，但这种表达的语义是模糊的且与范式要求相矛盾。若基于面向对象模型进行数据库逆向工程，需要解决关系模型中模式分解的表示问题。

(5)引用完整性：理想的引用完整性约束应由数据库系统强制而不是由每个应用的客户代码实现，但过去由于有些数据库系统并不支持引用完整性，因此情况并不总是如此。在进行数据库逆向工程时，需要解决关系模型引用完整性约束的表示问题。

(6)空值：关系数据库模式中对非空约束比较松，通常一个模式允许空值存在是因为在单个事务范围内不具有设置值的能力。一般有两种方式：一个逻辑应用的非空约束不可能总被说明，人们在使用时必须仔细地解释模式。实际中存在许多不同的策略，比如使用NULL、默认值、或使用一个不会在应用中出现的奇怪的值来表达空值。在进行数据库逆向工程时，需要解决关系模型空值表示问题。

(7)枚举域：有些关系数据库不支持枚举域的概念，这种对枚举域支持的缺乏可导致逆向工程的复杂化。一般问题包括：在数据库设计中经常会遇到枚举，缺乏对枚举类型的支持，必然导致理解上的困难。应用中经常存在不作解释的枚举值，这极大地妨碍了对语义的理解，在进行数据库逆向工程时，需要解决关系模型枚举域的表示问题。

3 基于EXPRESS面向对象模型的关系数据库逆向工程分析

我们采用EXPRESS标准的信息建模语言来进行数据库模型的描述，EXPRESS是一种形式化的模型描述语言，是组成STEP的STEP是ISO于九十年代推出的产品模型数据交换的国际标准的所有标准中一个核心的国际标准，是STEP所有实施方法和有关工具的基础。实体说明语句是EXPRESS语言的核心内容，它创建一个实体数据类型，用来描述一类具有共同特性和行为的现实世界中的物理或概念对象，对象的数据元素用属性来描述，而行为则通过静态约束来表示，具体实体说明的语法为：

```
<entity-decl>::=ENTITY <entity_id>[<subsuper>];'
    {<explicit_attribute>}[<member_function>]
    [<derived_clause>][<inverse_clause>]
    [<unique_clause>][<where_clause>]
    END_ENTITY';'
```

这里entity_id是实体标识符，subsuper的说明是实体说明中的重要部分，它描述了实体之间的父子关系，子类可以继承父类的全部特性和行为，其描述体现了面向对象技术中的继承机制，为建立复杂对象的信息模型提供了强有力的工具。

实体的属性分为显式属性和逆向属性二类。explicit_attr说明的属性是实体的基本属性，inverse_clause说明的是逆向属性，逆向属性的使用为建立两个实体之间所属的约束关系提供了一种非常简捷的方式。属性定义中的值域可以是基本数据类型，也可以是另一个实体类型，它体现了面向对象技术中的合成机制，进一步加了语言的建模能力。

unique_clause指明对于实体的某个或某些属性，其实例必须保持唯一性。where_clause指明了对实体值域的一些约束，只有属性值满足该值域规则约束的实例才属于该实体中的实体。

在对EXPRESS语言的核心实体描述的分析中，可以得出EXPRESS语言非常适合于用来描述关系数据库逆向工程，其基本实现原理为：

(1)支持实体及实体间继承的描述：EXPRESS语言的核心就是实体的描述能力，关系数据库模式中的表格可以表达为EXPRESS语言中的实体。对于从关系数据库中抽象出来的普化和特化关系(需要领域专家的参与)，可通过EXPRESS实体定义中的子类和超类说明来方便地表达。

(2)支持实体间的双向合成联系：EXPRESS语言的实体描述中支持对显式属性和逆向属性的说明。显式属性说明中可以正向说明两个或多个实体之间的联系，即本实体可以引用一个或多个其它实体；逆向属性则是定义本实体被另外实体引用的关系，即在另外的实体定义中存在显式属性，该显式属性的值域即为该实体类型。显式属性可以方便地从一个方向建立实体间的联系，逆向属性的使用为建立两实体之间的合成

关系提供了一种非常简捷的方式。对于关系数据库中可能出现的双向联系，用 EXPRESS 语言中的显式属性和逆向属性的说明可以有效地表达，且其语义比较清晰。

(3)支持唯一性定义：EXPRESS 语言实体定义中的唯一性规则，是用来限制该实体实例的基于值的唯一性，其作用相当于关系表中的主键。由于 EXPRESS 语言实体中唯一性规则中指定的属性可以是一个或多个属性，且其属性值可以为空也可以不为空，因此对于关系数据库表中的主键确定和标识问题，可用 EXPRESS 语言的唯一性约束规则类表达，且其语义清楚明了。

(4)支持多种关系：EXPRESS 语言实体关系的定义可支持实体间的多种关系，既支持一对一、一对多的关系，也支持多对多的关系，这可通过实体属性说明是单个实例引用，还是多个实例引用来表达，由于 EXPRESS 支持聚合数据类型，因此对一对多和多对多的关系表达非常直接。对于关系数据库模式中表间的多种实例关系，可用 EXPRESS 语言实体属性非常简单地进行说明。

(5)支持空值：EXPRESS 语言实体属性定义中有一个选项 OPTIONAL，若属性说明为可选的，则该实体实例对应属性的值可以为空，也可以不为空；若属性说明时没有 OPTIONAL 选项，则该实体实例的属性值不可为空。因此对于关系数据库模式中的许多空值或非空值的约束，可方便地通过 EXPRESS 语言实体属性说明的可选选项来确定。

(6)支持枚举类型：EXPRESS 语言支持枚举数据类型，因此对关系数据库模式中出现的枚举情况，可通过把它表达为 EXPRESS 实体属性定义中属性域为枚举类型来解决。

(7)支持各种约束：EXPRESS 语言的实体定义中有 WHERE 规则定义，WHERE 规则指明对实体值域的一些约束，只有属性值满足该值域规则约束的实例才属于该实体中的实例。因此对关系数据库模式中对表的各种约束，可通过把这些约束说明 EXPRESS 实体中的 WHERE 规则来实现。

4 基于 EXPRESS 面向对象模型的关系数据库逆向工程

一般来说，基于面向对象模型的关系数据库逆向工程主要包括两个阶段：一是数据结构的抽取阶段，二是数据结构的规范化阶段，图 1 是关系数据库逆向工程的一个演化过程图。

(1)数据结构的抽取：这个阶段发现完整的数据库模式，包括所有隐含的和显式说明的结构和约束，其处理过程为：①数据库 DDL 语句的分析：对包含在模式描述和应用程序中的数据结构说明语句的直接分析，它直接产生一个逻辑模式；②数据分析：这个过程用来分析表格和数据库的内容以确定数据结构和特性、测试假设，这个过程也可用来发现隐藏的和非说明性的结构；③模式集成：当同时处理多个数据资源时，分析人员可获得许多不同的抽取出来的模式，最后通过模式集成过程来集成所有模式以形成一个完整的逻辑模式，由于这时的模式是按数据库的特定模型来说明的，所以该逻辑模式仍然包含许多优化的和经过转换的结构，因此必须对该逻辑模式进行规范化处理得到规范化的概念模式。

(2)数据结构的规范化：这个阶段负责数据库模式的概念解释，包括检查和转换或清除非概念化结构、冗余、属于技术优化和依赖于具体数据库系统的结构，其处理过程为：①还原模式转换：逻辑模式是概念结构到数据库模型的技术转换，通过还原模式转换，分析人员可以标识这种转换踪迹并把它们替换成原来的概念结构。②还原模式优化：对逻辑模式进行检查以发现用于设计优化目的的结构，并把这些结构从逻辑模式中清除。③概念模式的规范化：这个过程重构基本的概念模式以使它具有所期望的一般概念模式的特征。如简单性，最小性、可读性、通用性、扩充性等。

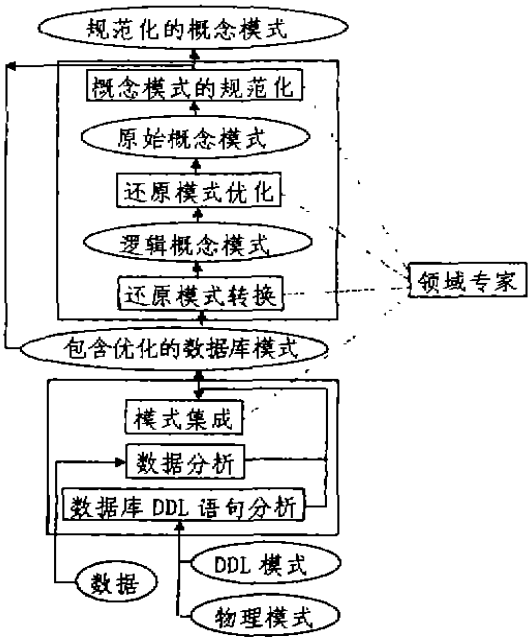

图 1 基于面向对象模型的关系数据库逆向工程处理过程

基于面向对象模型的关系数据库逆向工程中对数据结构的抽取不可能随意进行，由于关系数据库有其自己的数据对象定义和建立对象间联系的一些方法，

因此把一个关系数据库模式逆向设计为一个面对象数据库模式需要了解已有对象及其间的联系，其具体处理过程如下：

①建立一个初始对象模型，作为模型一部分定义的类可以通过审核数据库内的记录或关系模式中的表格而获得，记录或表格中包含的项成为类的属性。

②确定候选外关键字：检查属性以确定它们是否被用来指向另外一个记录或表格，那些用来作为指针的属性成为候选外关键字。这些候选关键字可用来建立面向对象模型的合成关系。

③细化临时类：在初始对象模型的基础上，通过领域专家的参与，确定在模型中是否存在相似的类可以合并成一个类的情况，若出现该情况则合并这些相似类成为一个类。

④定义普化：检查具有相似属性的类，以确定是否可以通过构造一个类层次并建立一个普化类作为类层次的头，这个步骤也需要有领域专家的参与。

⑤发现联系：采用相应技术，建立类与类之间的联系，这个步骤比较复杂，也需要有领域专家的参与。

应注意到关系数据库的逆向工程是一个需要大量知识的活动，在进行关系数据库逆向工程时不可能完全实现数据库模式的自动推导，因此在分析过程中需要增加领域专家对中间分析结果的判断，以帮助确定抽取的结构是否属于概念模式中的结构。同时我们认为一个较为完整的关系数据库逆向工程是一个循序渐进的、不断提高的分析过程，需要对逆向工程中产生的中间结果进行分析并考虑领域专家的意见。

参 考 文 献

1 Hainaut J-L. Database Reverse Engineering: Models, Techniques and Strategies. In Proc. of the 10th Conf. on Entity-Relationship Approach, San Mateo, E-R Institute, 1991

2 Hainaut J-L., et al. Transformational techniques for database reverse engineering. In Proc. of the 12th Conf. On Entity-Relationship Approach, Arlington-Dallas, E-R Institute, 1991

3 Hainaut J-L, et al. Database Design Recovery. In. Proc. Second Working Conference on Reverse Engineering, ACM SIGSOFT, 1995

4 Hainaut J-L, et al. Database Reverse Engineering from Requirements to CASE Tools. Journal of Automated Software Engineering, 1996, 3(1)

5 Prem. An Approach for Reverse Engineering of Relational Datbase. CACM, 1994, 37(5): 42～49

6 Roger H L, Reverse Engineering of Relational Databases: Extraction of an EER Model from a Relational Database, Data & Knowledge Engineering, December, 1994

7 ISO 10303-11: 1993. The EXPRESS Language Reference Manual[S]

8 Roger S. Pressman. , Software Engineering, A Practitoner's Approach, McGraw-Hill, 1997

1999 年《计算机科学》各项文献计量指标的评估结果

计算机学科	
总被引频次	396
影响因子	0.675
即年指标	0.088
自引总引比	0.16
地区分布数	23
基金和资助论文比例	0.56
海外作者论文数	4
指标综合加权评分	68.66

说明：1）上列数据由"中国科技信息研究所信息研究分所中心"提供（加盖公章），属科技部发展计划司委托项目之列；

2）1998 年《计算机科学》影响因子为 0.532，已进入 1286 种中国科技论文统计源期刊的前 100 名，居第 38 位，名列计算机类前茅；1999 年，其影响因子为 0.675，又提高了 26.9%；