

数据库设计工具集 DBTOOLS 的设计

杨冬青 唐世渭

(北京大学计算机科学技术系, 北京 100871)

DESIGN OF DBTOOLS: A DATABASE DESIGN TOOL SET

Yang Dongqing and Tang Shiwei

(Department of Computer Science and Technology, Peking University, Beijing 100871)

Abstract In this paper we present a database design tool set (DBTOOLS) which consists of three tools: Extended E-R model design tool (EERD), E-R model to relational model transformation tool (ERTR), and DFD, DD to relational model transformation tool (DDTR). DBTOOLS works in a combined mode of automatic transformation and human interference, and supports conceptual structure design and logical structure design of relational databases. In the design of DBTOOLS, we emphasize its support to semantic information description and utilization, and incremental development of database application systems.

摘要 本文介绍的数据库设计工具集 DBTOOLS 包括扩充的 E-R 模型设计工具 EERD、从 E-R 模型到关系模型转换工具 ERTR、从 DFD、DD 到关系模型转换工具 DDTR。DBTOOLS 以自动转换和人工干预相结合的方式工作, 提供对关系数据库概念结构设计和逻辑结构设计的支持。DBTOOLS 的设计中强调语义信息的描述和运用, 以及对数据库应用系统增量开发的支持。

§ 1. 两种数据库设计途径

大型综合数据库系统或数据库应用系统的开发过程中, 数据库设计是一个重要环节。在数据库设计实践中, 根据对数据库的需求和应用背景的不同, 通常采用的设计途径有两种。

如果数据库主要是用来综合组织不同来源的大量数据, 为各方面对信息的要求提供综合查询、统计报表等服务, 而不是作为直接支持应用系统运行的核心, 则在数据库设计中通常采用传统的数据库设计方法。即将数据库设计分为概念结构设计、逻辑结构设计、物理结构设计等几个主要阶段。在概念结构设计阶段中, 分析来自各个数据来源的各类数据对象、数据对象之间的联系、数据依赖、数据的完整性、安全性约束条件等, 用 E-R 图

以及附加的说明把它们表示出来,并且综合各个部分的 E-R 图形成整个数据库的全局 E-R 图.在逻辑结构设计阶段,将 E-R 模型转换为所选用的 DBMS 支持的数据模型,例如网状模型、关系模型等.在物理结构设计阶段,再加进对于数据的存储结构和存取方法的考虑,完成整个数据库设计过程.

如果数据库是用来作为支持应用系统运行的核心,数据库中存储的数据是应用系统在实际业务活动中收集加工的信息,并且是支持业务活动或支持管理、决策的基础,则数据库设计通常结合在整个应用软件系统设计的过程中.在软件需求分析阶段,分析处理需求的同时分析了数据在系统中的流程.应用系统的数据需求在软件需求分析阶段产生的文档“数据流程图”(DFD)、“数据字典”(DD)中得到充分反映.根据 DFD 和 DD 提供的信息,在应用系统的总体设计阶段进行数据库的逻辑结构设计.在这个阶段,往往也画出反映数据库概念结构的 E-R 图,但通常不用它作为转换到逻辑结构的基础,而是用它作为与用户交流的工具,使用户对设计中的数据库的结构有直观的了解,便于发表意见,改进和完善数据库的设计.同时 E-R 图也用来检验根据 DFD、DD 的信息设计的数据库逻辑结构是否完全包括了用户所需要的信息.

目前实际应用的数据库环境基本上都是关系型的,因此数据库逻辑结构设计的结果都是关系模式.规范化理论对于关系数据库的设计有重要指导意义.不论采用上述哪一种途径设计数据库,都要用规范化理论对产生的关系模式进行检验,以避免或减少数据库的操作异常.

§ 2. DBTOOLS 概述

为减轻数据库设计人员的劳动,提高数据库结构的合理性和数据库的应用效率,我们设计了一个数据库设计工具集 DBTOOLS.它包括三个工具:扩充的 E-R 模型设计工具 EERD、从 E-R 模型到关系模型转换工具 ERTR、从 DFD、DD 到关系模型转换工具 DDTR. DBTOOLS 提供对上述两种数据库设计途径的概念结构设计阶段和逻辑结构设计阶段的计算机辅助支持.

2.1 DBTOOLS 的主要功能

EERD 辅助数据库设计人员基于扩充的 E-R 模型建立数据库概念模型.它主要包括 E-R 图建立、修改和函数依赖、数据约束信息输入功能,视图集成功能,E-R 图“规范化”功能和设计文档输出功能.

ERTR 实现从全局 E-R 图到关系模式的转换和关系模式规范化.它主要包括 E-R 图到关系模式转换功能,关系模式规范化功能,根据语义信息产生数据库应用程序设计指导信息的功能,以及关系模式和应用程序设计指导信息输出功能.

DDTR 实现从软件需求分析产生的有关信息到关系模式的转换.它主要包括 DFD、DD 信息录入功能,根据 DFD、DD 信息转换生成关系模式的功能,根据 E-R 图检验关系模式完全性的功能,关系模式规范化功能,产生数据库应用程序设计指导信息的功能,以及关系模式和应用程序设计指导信息输出功能.

2.2 DBTOOLS 的系统模型和系统结构

DBTOOLS 的系统模型如图 1 所示. DBTOOLS 从数据库设计人员处接收关于 E-R 图、数据依赖、数据约束、DFD、DD 等信息,在视图集成、关系模式规范化的过程中与数据

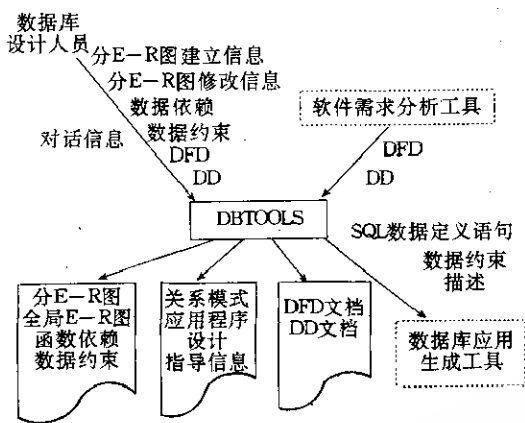


图1 DBTOOLS系统模型

库设计人员进行对话,将设计结果以文档形式输出. DBTOOLS 也可与其他软件工具接口,从软件需求分析工具接收 DFD、DD 信息,将生成的关系模式和应用程序设计指导信息以一定形式传递给数据库应用生成工具作为输入.

DBTOOLS 的核心是一个统一的数据字典,通过数据字典将数据库设计的不同阶段衔接起来,实现一个阶段的设计结果到下一个阶段的设计结果的部分自动转换;利用数据字典

中的信息进行设计结果的完全性、一致性检查;基于数据字典中的信息产生数据库设计文档和数据库应用程序设计指导信息.

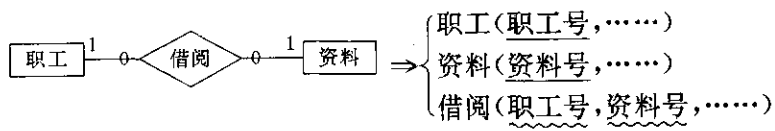
§ 3. 语义信息的描述和运用

DBTOOLS 支持数据库设计人员描述更丰富的语义,将语义信息运用到数据库设计过程中,并产生应用程序设计指导信息.

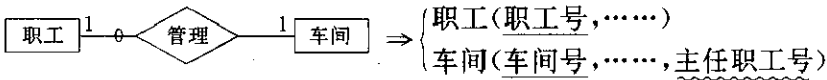
EERD 辅助数据库设计人员建立扩充的 E-R 模型. 除传统的 E-R 模型中采用的实体、属性、联系等数据抽象外,数据库设计人员还可以说明弱实体,实体集间的不交子集概括关系 (disjoint subset generalization), 相交子集概括关系 (overlaped subset generalization), 联系的强制性 (mandatory)、任意性 (optional), 以及属性间的函数依赖, 属性值的约束、复杂关联等, 从而将现实世界的语义更充分地反映在数据模型中. 这些语义信息对于 E-R 模型到关系模式和 DFD、DD 信息到关系模式的转换产生直接影响, 并且是数据库应用程序设计的重要指导.

例如, 两实体集间的一对一联系转换为关系模式时至少有三种不同的转换方法: 可以建立一个表示联系的关系, 可以在某一个参与联系的实体集所对应的关系中加上另一个实体集的码属性来表示此联系, 也可以在参与联系的两个实体集所对应的关系中都加上对方的码属性来表示此联系. 对于一个具体的一对一联系如何确定采用哪种转换方法呢? DBTOOLS 的扩充 E-R 模型中包括了关于联系的强制性的语义, 以此为据, ERTR 就可以选定最合适的转换方法了. 以实例说明如下: (在 E-R 图中, “0”表示联系是非强制性的, 不带“0”表示联系是强制性的.)

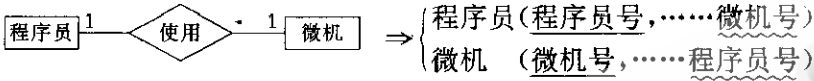
情况 1: 两方都是非强制性的(任意的)



情况 2:一方是强制性的

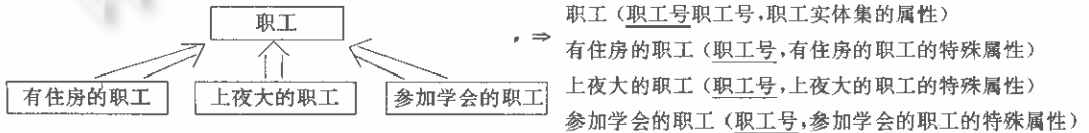


情况 3:两方都是强制性的



上面三种情况下转换得到的各关系模式中的外码(下加波浪线表示)均不允许取空值。

又如,若几个实体集间存在相交子集概括联系,则将在将实体集转换为关系时,不是简单地将各实体集的属性对应地转换为关系中的属性,而是要从各子实体集所对应的关系中去掉已包含在超实体集中的共同属性(除非数据库设计者出于效率考虑不同意去掉),只保留码属性和子实体集的特殊属性,如下例所示。



并且产生应用程序设计指导信息:“删除‘职工’关系中的元组,则同时删除其他三个关系中可能存在的相应元组”,“往‘有住房的职工’关系中插入元组,则首先检查‘职工’关系中是否存在相应元组”等。

此外,分析属性值的约束,属性间复杂关联等描述信息,可导出对表进行插入或更新时,同一元组的几个属性值必须满足一定的相关条件,或一个表中某元组的两个属性值之间的对应关系必须满足另一个表中给出的规定等这样一些复杂的程序设计指导信息.这些指导信息可以以文档的形式提供给应用编程人员参考,也可以经过转换成为数据库应用生成工具的输入信息。

§ 4. 交互式的视图集成和规范化

我们在 DBTOOLS 的设计中并不过分强调数据库设计过程的完全自动化,而是提供方便的界面给用户,使数据库设计人员能够充分表达他们的设计意图,把他们关于语义、效率等的考虑融合进计算机辅助的数据库设计过程中.我们的这一思想集中体现在 DBTOOLS 的视图集成和规范化功能中。

4.1 视图集成

EERD 的主要功能之一是将用户描述的多个分 E-R 图集成为全局 E-R 图,集成中考虑的两个主要问题是解决冲突和消除冗余。

在视图集成的过程中,DBTOOLS 的数据字典起着重要作用.EERD 基于数据字典中

记录的关于实体、属性、域、联系,以及它们相互关联的信息,对于不同视图中同名的对象做出可能同义或可能异义的初步判断,甚至对于不同视图中不同名的对象做出可能同义的初步判断,然后提请用户确认或否认,以便 EERD 作出合并或改名等处理.当 EERD 发现可能的冗余联系或可导出的联系时,也提请用户确认,然后消除掉.

4.2 规范化

DBTOOLS 将规范化理论运用到概念结构设计和逻辑结构设计两个阶段中.在概念结构设计阶段对 E-R 模型进行规范化,规范化以属性间的函数依赖为基础.除了数据库设计人员利用录入界面已经说明了的函数依赖外,通过 E-R 图还可推导出一些函数依赖,这些推导出的函数依赖也存储在数据字典中.

对 E-R 图的“规范化”包括对实体型的规范化和冗余联系的消除.对实体型的规范化指的是若一个实体型内各属性间的函数依赖关系违背 BCNF 的规定,则通过实体型的分解、实体型中多余属性的删除,或实体型间属性的转移等将其规范化.冗余联系的消除指的是若一个联系可由其他联系逻辑蕴含,则将它从 E-R 图中去掉.EERD 在此规范化过程中经常与数据库设计人员对话.一个实体集能否分解,一个联系是否真正由其他联系蕴含等,都要征求设计人员的意见.

关系模型的规范化以从 E-R 图或从 DFD、DD 转换得到的关系模式和数据字典中的各种关联信息为基础进行.规范化包括两个方面的工作.第一,逐个考察关系模式,对于不符合 BCNF 的关系模式,采用模式分解或删掉多余属性的办法提高其规范化程度.第二,综合考察所有的关系模式,删掉由其他关系模式蕴含的关系模式.规范化过程中保持与数据库设计人员的对话,充分尊重扩充的 E-R 模型所反映出的语义和数据库设计人员的决策,不作强制性的关系合并和分解.例如,超集和子集所对应的关系虽然码相同,但并不进行合并;当关系中由于有了一个冗余的属性而不符合 BCNF,但数据库设计人员考虑到处理的方便或效率问题而不同意删掉此属性时,尊重他的意见.

§ 5. 对数据库应用系统增量开发的支持

DBTOOLS 支持数据库模式的逐步扩展和数据库应用系统的增量开发.

由于人力物力或其他方面的原因,用户可能需要先分析、设计整个综合数据库的一个部分,以后再逐步加进其他部分,最后成为一个完整的综合数据库.用户甚至希望逐个子系统地开发一个大的数据库应用系统,即先设计数据库的一部分,开发在其支持下的应用程序,形成一个或几个子系统,并投入运行.以后再根据其他子系统的需要逐步扩充数据库模式,并逐步增加应用程序,最后形成完整的数据库应用系统.DBTOOLS 对上述设计和开发方式提供完全的支持,保证对数据库的扩充不影响已有的子系统的运行.

为正确支持上述各种背景下的数据库设计,EERD 工具的视图集成有三种不同的工作方式:第一,一般的视图集成,在考虑解决冲突、消除冗余时,各分 E-R 图处于同等地位.第二,数据库模式扩展时的视图集成,这时原有的 E-R 图是经过集成,用户认可的,在解决冲突、消除冗余时把它放在主导地位,新扩展的分 E-R 图向它靠拢.第三,数据库应用系统增量开发时的视图集成,这时原系统的 E-R 图是已运行的应用系统的基础,合成时,新开发的分 E-R 图只对原 E-R 图起扩充作用,不能引起修改.

相应地,针对数据库应用系统增量开发的情况,ERTR 工具中从 E-R 图到关系模式

转换时做了特殊考虑,转换以已开发系统的数据库模式为基础进行,对数据库模式的扩展只能是增加关系,原来的关系中增加属性,原有的属性列增加长度.这样,原来已运行的子系统的应用程序不必重新编制,数据库中已有的数据不重装,新增加的子系统可以共享数据库中已有的数据.

§ 6. 结束语

基于设计数据库和开发数据库应用系统的经验和体会,我们设计了数据库设计工具集 DBTOOLS. 我们很注重工具集的实用性,提供友好的、交互式的界面,使得数据库设计人员在计算机辅助的数据库设计过程中能充分表达自己的设计意图. 我们特别强调了在数据库概念结构设计阶段描述更丰富的语义,并把语义信息充分运用到数据库逻辑结构设计和对数据库应用程序设计的指导中,我们还从几个方面强调了对数据库应用系统增量开发的支持,因为增量开发方式是经常采用的一种应用系统开发方式.

参考文献

- 1 萨师煊、王珊,《数据库系统概论》(第二版),高等教育出版社,1991.
- 2 唐世渭、杨冬青、黄琨,DBIT——一个数据库集成工具,第九届全国数据库学术会议论文集,1990, 719—725.
- 3 杨美清、唐世渭、方裕、杨冬青,管理信息系统的开发途径——一个大型系统 BD-MIS 的研制,《软件学报》,第 2 卷第 1 期,1991, 1—11.
- 4 T. J. Teorey, Database Modelling and Design, Morgan Kaufmann Publishers, In, 1990.
- 5 T. J. Teorey, D. Yang, J. P. Fry, A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model, ACM Computing Surveys, 18, 2, 1986, 197—222.
- 6 J. D. Ullman, Principles of Database Systems(2nd Ed.), Computer Science Press, Inc., 1982.