第3卷第4期.
1992年11月

软 件 学 报
JOURNAL OF SOFTWARE

Vol. 3, No. 4
Nov., 1992

数据库应用程序的转换方法

杜小勇

(中国人民大学数据与知识工程研究所,北京 100872)

THE METHODS OF DATABASE PROGRAM CONVERSION

Du Xiaoyong

(*The Institute of Data and Knowledge Engineering, People's University of China, Beijing* 100872)

Abstract Database program conversion frequently appeared and widely researched in variant database fields. The paper summarized the methods of data base program conversion, which mainly include syntactical-based, semantical-based, and rule-based methods. It also emphatically discussed the efficient conversion methods used in the distributed database system DQS/SEIS.

摘要 数据库应用程序转换问题已经在广泛的领域内进行了研究.本文首先综述了该领域的研究,总结了各种转换方法并归纳为三类,即基于语法的方法,基于语义的方法和基于规则的方法,着重讨论了分布式数据库查询系统DQS/SEIS中所采用的转换方法.

§1. 引 言

数据库的操作方式有两类,一类是"即席"方式,一类是"嵌入"方式.所谓"嵌入"是指,数据库操作语句不是孤立地出现,而是嵌入到某种宿主语言中构成一段程序来执行,以完成对数据库的一系列存取,获取必要的信息,对于这种操作方式,当我们要考虑数据模型的转换时,就不能单纯地考虑数据库操作的转换而必须考虑整个数据库应用程序的转换问题.数据库应用程序的转换问题可以非形式地描述如下:已知(1)一个数据库模式(旧模式)以及运行在该模式之上的一段数据库应用程序;(2)一个数据库新模式及其数据操纵语言,如何将旧模式上的数据库应用程序转换成新模式上的应用程序,并使之"等价"地运行.关于"等价"的定义很重要,对不同的问题应采用不同的等价定义,有些要求较强,有些则可以较弱些,在此我们给出所谓"保持I/O等价"的定义,它是一种最基本的等价,在某些实际应用中,还可以定义更弱的"可接受的等价"[1].

记数据库系统是一个三元组:DBS=<S, D, P>,其中S是数据库模式,D是存贮的数据

本文1989年7月14日收到,1991年1月7日定稿.作者杜小勇,讲师,1988年硕士毕业于中国人民大学,主要研究领域是数据库,管理信息系统.

库，$P=\{P_1, P_2, \cdots, P_n\}$是一组数据库应用程序，每一个程序作用于D都将产生一个输出：$Pi(D)=0i \quad (i=1, 2, \cdots, n)$

设T是一个转换过程，作用于DBS，将产生一个新的数据库系统$DBS'=<S', D', P'>$而且$Pi'(D')=0i' \quad (i=1, 2, \cdots n)$

其中$Pi'=T(Pi)$, $(i=1, 2, \cdots, n)$, $S'=T(S)$, $D'=T(D)$

如果$0i' \equiv 0i$ $(i=1, 2, \cdots, n)$，则称T是保持I/O等价的.

数据库应用程序转换问题是一个很普遍的问题，在广泛的数据库领域中做过研究，比如数据库重构问题的研究，分布式数据库及多库系统的研究等. 本文将首先综述数据库应用程序的转换方法，然后结合实际系统分布式数据库查询系统DQS/SEIS的实现介绍其程序转换的具体方法.

§2. 数据库应用程序的转换方法

通过对各领域中提出的数据库应用程序转换方法的比较分析，可以将现有方法归为三类：

(1)基于语法的方法

这种方法对源程序中的每一条DML语句，用一段等价的，重构后的数据库上的DML组成的程序与之对应，当执行源程序时，每接收一条源DML时，就调用相应的一段程序来保持源数据库应用程序的行为，这种方法也称作DML仿真. 它又可进一步分为解释的方法和编译的方法，前者通过动态地为每一个源DML语句构成一段在目标库上运行的程序来保持源程序的行为，而后者则通过匹配并调用事先设计好的模板来完成源DML的任务. 基于语法的方法在处理复杂的数据结构时会变得很复杂，在这种情况下，转换软件需要估计每个DML在源数据库中的操作以决定当前状态，以便在新的数据库上执行等价的DML操作，此外还要维护两种数据库的结构描述，映射描述，状态指示字等.

(2)基于语义的方法[2-4]

这是讨论的最多的一种方法，它强调对数据和程序的语义描述，并采用一种与具体数据模型无关的抽象表示作为中间描述语言来实现程序的转换. 它一般需要两个过程，首先对源程序进行分析，找出程序存取数据库的语义，并用抽象的中间表示进行描述，这一过程称为分析阶段；然后从中间描述中推导出目标程序，包括目标库上的数据库操作语句和应用程序的重构，这一阶段称为综合阶段. 这种两阶段的方法，由于有一种抽象的中介描述，使得对源程序的分析表示，以及到目标程序的转换都更简单了. 中间抽象表示方法的设计对于这种方法的实现是至关重要的.

(3)基于规则的方法[5]

这是最近才提出的方法，它采用一个基于规则的转换器来实现语言间的转换. 这种基于规则的方法较之经典的算法方法具有很多优点，如容易写规则，规则可以不断地增加，规则可以用于双向转换，可从例子中自动生成规则等. 基于规则的方法适用于那些关系比较松散的分布式数据库或多库系统中的语言转换，例如IDA系统为了在多个不同的数据库上使用统一的语言存取信息，设计了一个数据库存取模块，作为从统一查询语言到各关系语言的转换器.

数据库应用程序转换的这三种方法可以说代表了转换器设计思想的三个阶段，但是每种方法都各有利弊，针对不同的应用要求，可以选用不同的方法来实现. 基于语法的方法以单个

DML 语句作为转换单位,转换算法简单而且易于实现,但很难对转换作优化,因而效率不高.基于语义的方法,程序的转换是基于对程序存取数据库的语义分析结果的,因此可以对目标程序作较多的优化,但系统花在分析上的开销很大,而且算法比较复杂.上述两种方法都是针对具体两种语言进行转换的,而且转换是定向不可逆的.当存在多种语言需要相互转换时,这样的转换器就会需要很多.第三种基于规则的方法,则对这种情况相当理想.因为通用的转换器适合于任何两种语言的转换,只要这两种语言转换能够用规则的形式表示并贮存在规则库中即可,但要形成完备的转换规则还比较困难,目前还没有见到比较好的方法.可见每种方法都各有利弊,在实际做一个转换器时,应根据具体情况加以选择或变通使用.

§3. DQS/SEIS 中数据库应用程序的转换方法

DQS/SEIS 是为我国国家信息系统建设而开发的一个基于大型机远程通讯环境的分布式数据库系统.它以 DL/I 作为底库,以已经建立的 DL/I 应用系统为其应用背景.由于在分布环境下,过程性的数据操纵语言(如 DL/I)会引起极大的通讯开销,因此我们为 DQS/SEIS 设计了一个多层次的系统结构,并在涉网层次上设计了一个具有说明性数据语言的层次式模型 SHM.关于 DQS/SEIS 系统结构及 SHM 的设计参见论文[6,7].这样在 DQS/SEIS 中就需要实现从 DL/I 程序到 SHM 的映射,以及反向映射.下面我们分别介绍这两种转换所采用的方法.

1. 从 DL/I 到 SHM 的转换

这实际上是要完成从导航操作到说明性操作的转换,显然,采用基于语法的方法是不行的,应当采用基于语义的方法.根据第二节的分析,我们首先要设计一种适当的中间表示形式,其次要找到一种方法来获取原应用程序对数据库存取的语义,并用中间表示形式描述出来,最后还要进行程序的综合工作,完成从中间表示到目标的转换,转换过程见图 1.

我们采用的中间表示叫做受限数据库,它是一个带有特殊形式限定条件的树状结构数据库,其限定条件的一般形式为 $Qs1 \wedge Qs2 \wedge \cdots \wedge Qsn$,其中 Qsi 是作用于片段 Si 上的简单条件的逻辑表达式,称为片段条件.简单条件就是形如:"字段变量 θ 常量"的关系表达式.从这里可以看出受限数据库的表达能力是有限的,如它不能用于表示关系代数查询等,但它用来表达 DL/I 的查询却是充分的.由于它在结构上和限定条件上和 DL/I 都十分相似,所以用它来描述 DL/I 应用程序对数据库的存取语义是比较合适的.

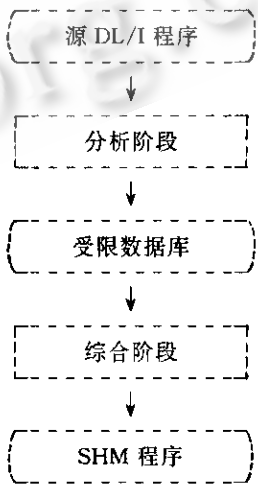

图 1　从 DL/I 到 SHM 的转换过程

对 DL/I 应用程序的分析,困难主要来自存在于记录间的一种次序(即层次序列),用户可以在程序中利用这个次序对数据库作存取,而且这种存取可以是毫无语义的,纯粹的存取.使用说明性操纵语言是无法表示这种存取的,因此要完成严格的保持 I/O 的等价是不可能的,但是,回想我们引入 SHM 目的,是要避免在全局级出现过

程性的 DL/I 操纵，减少通讯开销，因此我们可以不采用严格的保持 I/O 等价，而采用一种变通的方法. 首先，在分析阶段，利用间隔分析技术分析 DL/I 应用程序，根据语句的“序相关”性进行分组，然后对每一个“序相关”组用一个 SHM 操作去覆盖它们，这时的结果是原结果的超集，这里的“序相关 ”是指某个 DL/I 操纵语句所确定的记录的位置对后继 DL/I 语句的执行效果有直接影响(如 DL/I 的 GN 语句和它之前的任何一个 GET 语句都是序相关的)，最后再让原程序作用在这个超集上以获取最后的结果. 可见程序的转换实际上是要获得一个过滤器，它使得原数据库中那些对结果无贡献的数据尽可能地在本地过滤掉，然后将结果回送到查询发出场地，由后处理程序再作一次处理，最后得到查询结果(图 2).

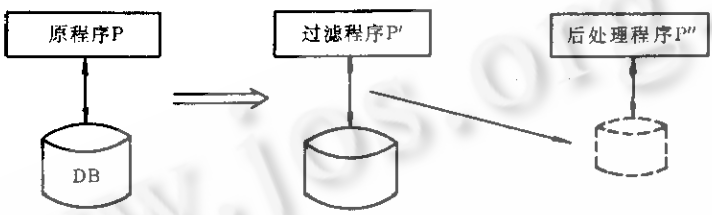

图 2 DQS/SEIS 中程序转换的基本思想

程序分析阶段包括以下三个步骤：第一步：寻找程序的循环体. 应用间隔分析技术，从程序流图中识别出全部程序循环体，包括嵌套循环体的复杂结构；第二步：识别 GET 语句间的位置依赖. 对程序流图中的 GET 语句实施图标记收缩过程获得具有位置依赖关系的语句集；第三步：为每个这样的语句集生成受限数据库表示.

程序综合阶段的任务包括以下二个步骤：第一步：为每个受限数据库生成对应的 SHM 语句，第二步：将 SHM 语句嵌入主程序中，并将原来对 DL/I 库的操作语句修改成对中间文件的操作，更重要的也是更困难的是要将原主程序中对 DL/I 状态信息及数据库工作区的值的引用修改成对文件存取信息和文件工作区的值的引用. 这样修改的程序就构成了图 2 中的后处理程序 P″.

在 DQS/SEIS 中，通讯系统采用的是前置机方案，主要的性能瓶颈是通讯联结次数，我们的转换方法在减少通讯联结次数上是十分有效的，尽管这种方法会增加一些数据在网上的传输量.

2. 从 SHM 到 DL/I 的转换

一个 SHM 查询，最终必须在各局部 DL/I 数据库上执行，这时就要进行从 SHM 到 DL/I 的转换. 我们采用基于语法的方法，而且为了提高查询执行的效率，采用模板匹配技术，即事先设计一组模板，这些模板应当具有通用性，它和具体的应用库结构无关. 当执行 SHM 语句时，根据 SHM 中目标片段和条件表达式中所涉及的片段形成一个该 SHM 语句的“查询图”，然后由这个“查询图”进行模板的匹配和调用. 具体包括以下三个方面：

(1)模板设计. 在 DL/I 数据库中，所有的查询都必须从根片段开始，所以记录的检索路径是从根片段开始到目标片段为止的线性树. 再由 DL/I 数据库关于片段层数的限制，我们可以设计出 15 个通用的模板，并用模板中的片段数作为模板的标识.

(2)模板匹配. 一个 SHM 的查询图通常是一棵树，为了能使用模板，需要将其分解成一系

列的带根线性树,然后以每个带根线性树为单位(称为单支查询)计算其片段数,并以此作为模板标识调用相应的模板.调用时以分布式数据库局部底库的PSB名,该线性树中各片段的片段名及限制条件作为参数代入模板.模板执行结果将获得一些中间结果记录,每个片段值都带有相应片段的片段号.

(3)单支查询归并.由于我们规定片段之间的逻辑关系是"与",因此各个单支查询结果应当作"与"归并,即如果两个单支查询在共同的祖先片段上有相同的值,那么应当将这两个单支查询在不同片段上的值归并到同一个祖先片段值上,如果在共同的祖先片段上仅有一个单支查询有这个值而另一个单支查询无这个值,那么应抛弃这个单支查询的结果记录,这样两两归并最后即可得到SHM语句的完整查询结果.

从DQS/SEIS应用程序转换的实例中可以看出:第一,数据库应用程序转换是很复杂的,必须认真对待;第二,尽管转换方法有基于语法,基于语义和基于规则三类,但对具体问题应灵活选择,变通使用;第三,对具体转换问题可采用特定的可接受的转换等价条件以便简化转换的算法.

§4. 结束语

数据库应用程序转换问题具有十分重要的实际意义,经过广大学者十多年的研究,现在已比较成熟,关键的问题是如何在实际问题中正确地选择并能够根据具体问题作出灵活的变通,本文对这一领域作了简要的总结,将转换方法归结为三类,并讨论了每类方法的特点.这对于实际系统的研制者在实现一个转换器时正确地选择和设计算法是很有意义的.这样的分析也指导了我们在实现DQS/SEIS系统时正确地设计其中的语言转换器.

作者感谢萨师煊教授,王珊教授的指导.

参考文献

[1]B. Shneiderman et al., "An Architecture for Automatic Relational Database System Conversion", ACM-TODS, Vol. 7, No. 2, 1982.

[2]R. H. Katz et al., "Decompiling CODASYL DML into Relational Queries", ACM-TODS, Vol. 7, No. 1, 1982.

[3]R. W. Tayer et al., "Database Program Conversion", Proceeding of 5th VLDB Conference, 1979.

[4]B. Demo, "Program Analysis for Conversion from a Navigation to a Specification Database Interface", Proceeding of 9th VLDB Conf. 1983.

[5]G. Piatetsky-Shapiro, "An Intermedia Database Language and Its Rulebased Transformation to a Different Database Language", Data and Knowledge Engineering, Vol. 2, No. 1, 1989.

[6]杜小勇等,分布式数据库查询系统DQS/SEIS设计实现中的若干问题的探讨,微型计算机,No. 3, 1989.

[7]Du Xiaoyong et al., The Global Conceptual Model Design in Distributed Databse Query System DQS/SEIS, Proceeding PPCC-3 Conf. 1989.