

Oracle 工程数据库中的面向对象设计技术

袁海斌¹, 袁海文¹, 张 新²

(1. 北京航空航天大学, 北京 100083; 2. 精詮资讯(上海)有限公司, 上海 200040)

摘 要: 数据库技术在工程领域的应用越来越广泛。同传统数据库比较, 对工程数据库则有许多新的要求, 如能够完成强大的建模能力和高效的面向对象的处理能力, 能满足网络化和实时分布式处理能力。结合面向对象技术的关系数据库是对传统事务处理数据库的扩展, 利用面向对象技术建立数据库的模型, 可以在满足工程数据库的要求基础上减少创建应用的过程, 是下一代数据库的一个研究方向。以主流数据库 Oracle 8i 为主, 讨论了面向对象的技术特点, 最后以实际工程数据库的对象建模为例作了阐述。

关键词: 工程数据库; 面向对象; Oracle 8i 数据库

中图法分类号: TP311

文献标识码: A

文章编号: 1001-3695(2003)08-0142-03

Object-Oriented Design Technology and Its Application in Oracle Engineer Database

YUAN Hai-bin¹, YUAN Hai-wen¹, ZHANG Xin²

(1. Beijing University of Aeronautics & Astronautics, Beijing 100083, China; 2. Systex Information(Shanghai) Co., Ltd., Shanghai 200040, China)

Abstract: Database technology is used widely within engineering field, new requirement is need in the engineer database compared with conventional database, for example, strong capability in module creation and high efficiency in object-oriented process is required in order to meet networked and real-time distributed processing. Relational database integrated with object-oriented technology is extension for traditional data transaction, by using object-oriented technology in database creation, application procedure can be minimized and engineering database requirement can be obtained. Object oriented technology is a new approach for the next generation database. This paper discusses the character of object-oriented technology based on Oracle 8i database, at last, the method to use object-oriented database is introduced and the example of engineer database module creation is given.

Key words: Engineering Database; Object-Oriented; Oracle 8i Database

1 序言

数据库技术的发展, 已经成为现代信息技术的重要组成部分, 而信息技术的发展, 又对数据库提出了更高的要求: 必须支持更多的用户和更大量的数据, 同时要能够处理数据流和非数据流等数据类型。总之, 数据库要有更贴近真实、完整的现实世界的特点。传统的数据库技术无论从网状、层次和关系数据库来看, 都不能完全地表达客观的数据对象, 因此, 必须要将传统数据库技术和面向对象的技术相结合, 以实现数据库功能的扩展和对未来工程数据库的有力支持, 这也正是基于工程数据库的特别要求。

2 工程数据库和面向对象技术

工程数据库是作为某种特定的工程应用而建立的数据库。目前基于工程数据库的应用很多, 如结合人工智能控制和故障诊断技术的专家系统就使用知识库和规则库, 而在现代化生产企业, 基于 CIMS 下的 CAD/CAM

系统完成零件模型从设计到组装生产, 其数据库所涉及的许多信息既有数值等信息, 更多的是图片等非数值信息。现代计算机仿真技术也使用庞大的数据库来运用声音、图像、动画等描述物体的状态, 同时要求有分布式实时的处理功能。总之, 对上述类型的工程数据库, 要求它们有复杂多变的数据类型和非数据类型的描述能力, 有高速、大容量的存储能力和高效率的分布式、实时、并发性的访问控制。因此, 采用面向对象的数据库管理(OODBMS)技术是一个必然的方向。面向对象技术的数据库实际上是传统的关系数据库的扩展, 正是它具有了重载、继承、多态等面向对象的特点, 因而能够描述对象间的复杂关系, 更贴近现实情况。采用面向对象技术增强了数据库的可扩展技术, 特别是在专家系统等数据库的数据增加非常迅速的应用领域, 均可以通过定义新的数据类型和操作来完成。

面向对象数据库在传统关系数据库基础上采用了与面向对象的程序设计相似的技术, 数据库中的对象是数据模型的一种抽象表示, 是相关数据的集合, 具有属性和方法。属性是指对象的组成、特征和状态, 而方法则用来描述对象的行为。对象的状态改变是通过方法来改变的, 也就是通过方法进行数据库内数据的操作。

对象具有封装性,每个对象都拥有自己的属性和方法,对象之间的交互通过方法来实现。类是对象的模板,对象则是类的具体实现;对象的子集为子类,子类可以继承父类的属性和方法,同时增加自己的属性和方法,但子类不继承父类的数据。继承增强了代码的可重用性,降低了应用开发的代价。面向对象的数据库模型能够充分表达实际工厂领域内事物对象和它们之间的相互联系,目前,采用面向对象技术的数据库已经越来越多地应用到各个工程领域中。而在 Oracle 8i 的版本中,已经加入了非常丰富的对象数据库设计技术。

3 Oracle 8i 数据库中的面向对象技术

Oracle 数据库是使用非常广泛的关系型数据库系统。在 Oracle 8i 中加入了面向对象的设计技术后,数据库的功能得到了扩充,与常规的数据处理一样,Oracle 能够完全处理和检索对象数据。所有能够使用在常规数据库上的功能,都能够在 Oracle 对象数据库上得到实现,检索、存储、触发器、备份和恢复等。

3.1 对象的建立

Oracle 数据库中,对象是包含数据类型和存储过程或函数的实体;数据类型是对象属性,存储过程或函数是对象方法的描述。对象类型的建立有三种方法:

(1) 实现用户定义数据类型,可以不包含方法,适用于不知道对象类型的具体形态时,是一种不完整的对象类型建立;

(2) 在关系数据库中创建嵌套表,用于结合关系型数据类型和面向对象数据类型;

(3) 创建对象表,实现完全的面向对象的数据库设计。

3.2 对象的表示和创建

在 Oracle 数据库中,对象是存储在数据字典中的,因此,可以通过查询数据字典来显示对象之间的相互关系。每个对象类型可以有一个或多个成员方法。对象类型的成员方法只是依赖于对象类型自身的存储过程或者函数,对象类型成员方法不能插入、修改或者删除数据库表中的信息。Oracle 数据库中提供了行对象和列对象,行对象可以看成是在关系数据库中建立的嵌套表,列对象是出现在表的一列或者作为其它对象属性。对象具有对象标识(OID),对象标识是和行对象相联系的,同时也用来进行对象间的关系运算,它是一种内置功能。在面向对象的数据库中同样也可以创建对象视图,通过对象视图可以保持在数据安全性的前提下完成数据维护。对象类型的创建需要有类型声明和类型体,类型声明是用于操作类型的公共接口,对象类型的类型体是类型方法的私有实现,用于实现对象的方法。

3.3 对象方法的建立

在 Oracle 数据库中,方法是一个 PL/SQL 过程或者函数,与对象类型同时存储,完成对象类型的封装。但又与存储过程的执行不同,方法是通过相关类型的对象而激活的,同时方法对相关的对象类型有完全的访问

权。构造方法、成员方法、映射方法(Map)和排序方法(Order)是主要的对象方法。对象类型的构造方法实际上类似于在关系数据库中的数据插入、删除等操作。对象类型的成员方法则是通过某种存储过程或者函数来处理对象的类型,该成员方法只依赖于对象类型自身。显然,对象类型成员方法不能完成插入、删除、修改等数据库表的操作。Map 和 Order 是 Oracle 中专有的方法,其实质是用来建立两个不同对象之间比较关系的一种方法,映射方法用于对数据库对象类型进行排序,对象类型的映射方法无需输入参数,返回一个标量数据类型;而排序方法通过接收输入参数返回方法结果。没有声明映射或者排序方法的对象类型只能完成相等或不等的比较操作。

4 面向对象的工程数据库在故障诊断系统的实现

基于专家系统的故障诊断是依靠知识库和规则库,经过推理,得到诊断结果。随着应用系统的运行,不断会有新数据的加入和已有数据的修改,因此用面向对象的技术设计数据库可以提高系统的扩展能力和运行效率,并且在技术开发上思路清晰。本文主要在于论述 Oracle 数据库面向对象技术的应用,因此不准备讨论更多的专业知识。下面从文献[3]提取一个例子作为 Oracle 数据库面向对象技术的应用实践,并做了适当的简化。数据的存储结构如图 1 所示,图 2 则为它们的对象关系。在图 2 中,列出了一个诊断对象数据结构 Diag_ Obj 和两个对象类型 Usr_ Type 和 Samp_ Type,后两个因为没有方法所以是不完全对象类型。

Diag_ID	Diag_LID	Diag_Addr	Diag_Data	Diag_Usr
Number	Number	Text	Object Type	Object Type
NUMBER	NUMBER	VARCHAR2(20)	Samp_Data	Usr_Type
PK				FK
MEMBER FUNCTION Diag_Match RETURN VARCHAR2				
MEMBER FUNCTION Diag_Examp RETURN NUMBER				
MEMBER FUNCTION Diag_Redo RETURN VARCHAR2				

图 1 诊断数据库的对象关系表示

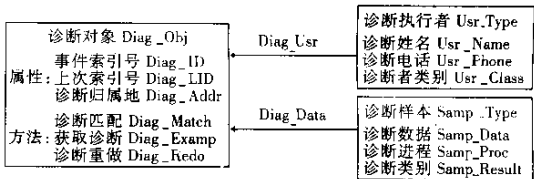


图 2 面向对象关系的诊断数据库数据结构

(1) 对象的创建

```
CREATE TYPE Usr_type AS OBJECT(  
  Usr_Name VARCHAR( 20 ),  
  Usr_Phone VARCHAR( 10 ),  
  Usr_Class NUMBER  
)  
CREATE TYPE Samp_Type AS OBJECT(  
  Samp_Data VARCHAR( 20 ),  
  Samp_Proc VARCHAR( 10 ),  
  Samp_Result NUMBER  
)
```

这种建立方法实际上是一种构造方法,具体的建立方法可以在参考手册中找到^[2]。在建立诊断对象时,诊断样本和诊断者可以嵌入到诊断对象中,从而实现成员方法建立对象。

```
CREATE TYPE Diag_Obj AS OBJECT(  
  Diag_ID NUMBER(  
    10, 0) NOT NULL,  
  Diag_LID NUMBER(  
    10, 0) NOT NULL,  
  Diag_Addr VARCHAR2(  
    20) NOT NULL,  
  Diag_Data Usr_Type,  
  Diag_Usr Samp_Type,  
  MEMBER FUNCTION Match  
    RETURN VARCHAR2,  
  MEMBER FUNCTION Examp  
    RETURN NUMBER,  
  MEMBER FUNCTION Redo  
    RETURN VARCHAR2  
)
```

```

Diag_ ID NUMBER ,
Diag_ LID NUMBER ,
Diag_ Addr VARCHAR( 20 ) ,
Diag_ Ustr Ustr_Type ,
Diag_ Data Samp_Type ,
MAP MEMBER FUNCTION Diag_ Index RETURN VARCHAR2 ,
MEMBER FUNCTION Diag_ Match RETURN VARCHAR2 ,
MEMBER FUNCTION Diag_ Examp RETURN NUMBER ,
MEMBER FUNCTION Diag_ Redo RETURN VARCHAR2
)

```

法：在对象类型体的对象内部完成对象类型的成员方法：

```

CREATE TYPE BODY Diag_ Obj AS(
MEMBER FUNCTION Diag_ Examp RETURN NUMBER IS
BEGIN
RETURN Diag_ LID + 1 ;
END ;
MEMBER FUNCTION Diag_ Match RETURN VARCHAR2 IS
BEGIN
...
END ;
MEMBER FUNCTION Diag_ Redo Return VARCHAR2 IS
BEGIN
...
END ;
);

```

(2) 对象表创建和对象数据结构操作

对象表的创建不同于关系数据表的创建,因为对象表中有嵌入的行对象,如上例所示,创建 Diag_ Obj 对象表时,使用如下方式诸如 CREATE TABLE Diag_ Obj_ Table of Diag_ Obj 就可以完成。在 Oracle 中,对象的操作通过 DML 语言进行操作,但是在引入对象概念后,就有些不同,以给 Diag_ Obj_ Table 中插入数据为例:

```

INSERT INTO Diag_ Obj_ Table VALUES(
Diag_ Obj( 2041 2022 'Local-Site' ,
Ustr_Type( 'ZhanYu' , '01066882200' 5 ) ,
Samp_Type( '0X80H' , 'final' , 'finished' ) )

```

当对象表含有对象引用属性时,必须使用 REF 或 VALUES 函数来处理对象的应用,REF 返回特定对象的对象标识符的引用或指针,同时在使用对象的属性或者方法时,使用点运算符来完成。

```

INSERT INTO Diag_ Obj_ Table
SELECT 2050 2026 'Remote-Site' ,REF( C ) ,Samp_Type( '0X80H' , 'fi-
nal' , 'normal' )
FROM Ustr_Type_ Table C
WHERE C. Ustr_ Name = 'Wei Hong' ;
UPDATE Diag_ Obj_ Table

```

```

SET Diag_ LID = Diag_ Obj. Diag_ Examp
WHERE Diag_ Addr = 'workshop' ;

```

通常可以在 SQL 会话中完成对象的操作,但是更常用的是使用 PL/SQL 过程完成 Select ,Update ,Update 和触发器等操作。

(3) 对象视图

在 Oracle 数据库中,视图也从关系型扩展到面向对象的类型,对象视图特指含有行对象的视图。如果以前是采用基于关系的数据库模型建立应用系统,采用对象视图就可以在不改变原有的系统数据结构而转变到面向对象结构的系统数据结构。这对于非常庞大的工程数据库来说,是一个很好的解决办法。在对象视图中同样也可以定义行对象和列对象,只不过在对象视图中,行对象和列对象是从关系数据库中合成的,视图对象是直接在服务器端完成 SQL 查询的,因此,可以减少查询过程和降低网络流量。所有的方法和本文上面所述的相似,可以参考相关的文献。

5 结束语

通过在 Oracle 数据库中采用面向对象的技术来设计工程数据库可以满足大数据量、分布式实时控制的要求,同时数据库很容易修改和扩充。然而在目前还没有关于面向对象的数据库的统一技术标准,现有的面向对象的数据库技术都是借用了程序设计语言技术。无论如何,面向对象的设计是一个发展方向,必将随着工程数据库发展而发展。

参考文献:

- [1] [美] Steve Bobrowski. Oracle 8 体系结构[M]. 王炎,等. 北京:机械工业出版社,2000.
- [2] Oracle 8i Application Developer's Guide - Fundamentals[M]. Oracle Corporation,1999.
- [3] 李立干. 基于 Internet 的数控系统故障诊断研究[D]. 北京:北京航空航天大学博士学位论文,2000.

作者简介:

袁海斌(1971-),男,江苏镇江人,在读博士生,研究方向为计算机过程控制与测量、现场总线应用研究、数据库工程应用。

并行与分布式计算、应用与技术(PDCAT '03)国际会议 征文通知

“第4届并行与分布式计算、应用与技术(PDCAT '03)国际会议将于2003年8月27日~29日召开,地点为中国四川成都。会议举办单位为美国IEEE、日本IEICE、英国IEE、国家自然科学基金会(NSFC)和西南交通大学(SWJTU),IEEE国际会议批准编号为#8948(参见www.ieee.org会议查询),会议英文论文集将由国际知名出版社公开出版。本次会议将在网络与通信、并行与分布式系统体系结构、并行与分布式处理算法与应用等方面,为世界范围内计算机与通信领域的科学家和工程技术人员提供一个发表最新成果的论坛。本国际会议开始于2000年香港的PDCAT '00,2001年在台湾举行PDCAT '01,2002年9月在日本举行PDCAT '02,2003年将在中国成都举行PDCAT '03。PDCAT '03国际会议论文截稿日期为2003年3月15日。会议详情请参见:
<http://home.swjtu.edu.cn/~imc/pdcat03/>。

联系人:西南交通大学计算机与通信工程学院院长 范平志 教授

联系电话(028)87600743 E-mail:p.fan@ieee.org <http://home.swjtu.edu.cn/~imc/pzfan/>