

# 面向对象数据库发展和研究<sup>\*</sup>

王功明<sup>1,2</sup>, 关永<sup>1</sup>, 赵春江<sup>2</sup>, 王蕊<sup>1</sup>

(1 首都师范大学 信息工程学院, 北京 100037; 2 国家农业信息化工程技术研究中心, 北京 100089)

**摘要:** 面向对象数据库作为第三代数据库具有前两代数据库无法比拟的优点, 满足复杂数据结构和海量存储需要, 是新型数据库如多媒体数据库、空间数据库、演绎数据库、工程数据库实现的基础。通过对比传统数据库的不足来说明面向对象数据库的优点, 讲述面向对象数据库中的七种关键技术: 数据库转换、模式演进、索引、事务管理、视图实现、版本管理、安全建模, 介绍面向对象数据库的几种实现方法和发展前景。

**关键词:** 面向对象; 数据库转换; 模式演进; 事务管理; 版本管理; 安全建模

**中图法分类号:** TP392      **文献标识码:** A      **文章编号:** 1001-3695(2006)03-0001-04

## Development and Research of OODB

WANG Gongming<sup>1,2</sup>, GUAN Yong<sup>1</sup>, ZHAO Chunjiang<sup>2</sup>, WANG Rui<sup>1</sup>

(1. School of Information Engineering Capital Normal University, Beijing 100037, China; 2. National Engineering Research Center for Information Technology in Agriculture Beijing 100089, China)

**Abstract:** OODBS that is called the third DB has excellent advantages compared with the former two generation DB. It satisfies with demand of complicated data structure and mass storage, which is the base of designing some new DB such as multimedia DB, space DB, deduction DB, engineering DB and so on. This paper first shows the merit of OODBS through comparing the deficiency of traditional DB, secondly introduces seven key technology of OODBS: DB switch, model evolution, index, affair management, view realization, edition management, safe modeling, finally presents several constructing methods of OODBS and expects the future of OODBS.

**Key words:** OO; DB Switch; Model Evolution; Affair Management; Edition Management; Safe Modeling

数据库发展经历了三个阶段。第一阶段是层次和网状数据库, 过程化程度较高, 一般用户使用困难; 第二阶段是关系数据库(RDB), 它以关系演算和关系代数为其数学基础, 以二维表为其数据结构, 利用非过程化数据操纵语言进行数据库管理, 采用内外概念模式的三层模式结构, 具有较高数据独立性, 成为20世纪70年代到80年代中期的主流数据库。上述层次、网状和关系数据库尽管设计和控制方式不同, 但都用于一般事务处理, 统称为传统数据库。近年来, 随着网络技术、多媒体技术、空间信息科学、信息管理、人工智能、软件工程和数据挖掘技术等领域的发展及新的社会需求出现, 信息无论从数量上还是结构上都远远超出了传统数据库能承受的范围。为了适应海量信息和复杂数据处理要求, 新一代数据库应运而生, 它们结合特定应用领域, 分为多媒体数据库(结合多媒体技术)、空间数据库(结合空间信息学和GIS)、演绎数据库(结合人工智能)、工程数据库(结合软件工程)等。与传统数据库相比, 它们既具有多样性(学科交叉的必然结果), 又有统一性, 建立它们的主要目的是为了处理海量信息和复杂数据结构。因此面向对象技术必不可少, 它们之间关系如图1所示。

## 1 面向对象数据库发展

### 1.1 面向对象数据特点

Peter Coad和Edward Yourdon这样描述面向对象: 面向对象=对象+分类+继承+消息。其中对象指一组属性及这组属性上专用操作的封装体。类是一组具有相同属性和操作的对象描述。继承是类之间一种基本关系, 指某个类的层次关联中不同类共享属性和操作的机制。消息是对象间通信的手段, 一个对象通过向另一个对象发送消息来请求其服务。此外, 面向对象数据特点还有封装、信息隐蔽、消息传递、多态性等。

综上所述, 面向对象数据与现实世界实体对象一一对应, 具有传统数据库数据不具有的两大特性, 即内容海量性和结构复杂性, 它们是构建新型数据库的基础。

### 1.2 传统数据库局限性

(1) 不能表示客观世界复杂对象。采用二维表表示数据及其关系, 语义表示能力差, 无法表示客观世界复杂对象, 不能揭示数据之间深层含义和内在联系, 缺乏数据抽象。

(2) 缺少对复杂数据类型支持。只能理解、存储和处理简单数据类型, 不能根据客户需要动态扩大数据集。碰到复杂问题常利用高级程序设计语言构造相应数据类型和操作, 既加重用户负担, 又不能保证数据一致性。

(3) 数据结构不能与行为相关联。对象有两方面内容, 即结构和行为。传统DB把前者映射到数据库模式中, 对后者没

收稿日期: 2005-07-04; 修返日期: 2005-08-24

基金项目: 国家“863”计划资助项目(2002AA714023); 首都师范大学研究生精品课建设项目(05531451)

有很好实现。

(4) 阻抗失配和语义断层, 不能与高级程序设计语言无缝集成。传统 DB 开发需同时使用数据库语言 (SQL) 和高级程序设计语言, 涉及模式和结构转换问题, 既容易丢失原数据结构语义, 又妨碍其他工具和用户在原有语义层次上共享数据。

(5) 不能主动检查和处理事件。传统 DB 管理系统是响应型, 被动接收客户输入, 不能主动与客户交互。

(6) 缺乏管理知识和对象的能力。传统 DB 处理对象是确定的、现存的, 不能很好地处理和管理实际应用中的二义性、未知对象。此外, 它们没有演绎和推理功能, 不能很好地管理知识, 无法满足 MIS、DSS、OA 和 AI 等领域进行高层管理和决策的要求。

(7) 缺乏对长事务和多重嵌套事务的响应和处理能力。传统 DB 只支持非嵌套事务, 对长事务响应较慢, 并且事务发生故障时恢复比较困难。

(8) 不能满足巨型数据库应用需要。随着多媒体技术、空间信息科学和数据挖掘技术等学科的蓬勃兴起, 处理的海量数据已非一般二维表可存储和管理, 而且数据结构越来越复杂, 有的还有语义动作, 使传统数据库显得力不从心。

综上所述, 传统数据库已不能满足复杂的实际应用需要。随着面向对象研究的深入, 把面向对象设计方法和数据库技术结合形成新一代数据库系统——面向对象数据库系统, 不仅是数据库学科发展需要, 也是推进计算机其他分支健康发展的必然结果。

### 1.3 面向对象数据库概念

面向对象数据库系统 (OODBS) 支持定义和操作 OODB 应满足两个标准: 首先它是数据库系统, 其次它也是面向对象系统。第一个标准即作为数据库系统应具备的能力 (持久性、事务管理、并发控制、恢复、查询、版本管理、完整性、安全性)。第二个标准就是要求面向对象数据库充分支持完整的面向对象 (OO) 概念和控制机制。

综上所述, 我们将面向对象数据库简称为: 面向对象数据库 = 面向对象系统 + 数据库能力。

## 2 面向对象数据库基本技术

### 2.1 数据库转换技术

异构数据库系统中各数据库模式和操作之间转换是一个关键研究课题。由于关系数据库系统主宰当今数据库应用领域, 而面向对象数据库能满足更高一级数据库要求, 所以有必要在这两种数据库模型中建立一种映射关系, 实现模式和操作相互转换。

转换一般有两种途径: 从关系 DB 到面向对象 DB (RDB to OODB) 和从面向对象 DB 到关系 DB (OODB to RDB)。本文以 OODB to RDB 为例介绍转换技术。转换时要保证一致性 (对象语义和动作信息在转换过程中不丢失)。转换包括数据模式和数据操作转换。

#### 2.1.1 数据模式转换

对象标识符是对象存在的唯一标志, 两个对象相等等价于其标识符相同。与关系模式不同的是面向对象中类属性分为原子属性、组合属性和集合属性。

数据模式转换指从 OODB 到 RDB 数据描述语言 (DDL) 的

转换, 基本思路是把父类属性扩展到所有子类中, 每个类映射为一个关系; 类的每个属性映射为它对应的关系属性。

类中不同类型属性作不同处理。默认对象标识符属性映射为 RDB 关键字属性, 原子属性映射为固定属性, 组合属性映射为与主属性对应关系关键字相关的外关键字, 集合属性映射为原子属性加上具有两个属性的关系, 其中一个属性是设置与对应的集合属性的联系; 另一个属性是处理集合元素。

方法转换是数据模式转换的重要转换, 方法有定义和调用。标准 RDB 无支持用户自定义函数和过程的机制, 近年来一些商业化 RDBMS 提供这方面的功能, 称为 PSM 子程序 (包含用户自定义函数和过程)。标准 PSM 子程序至少支持以下两种功能: ① 创建用户自定义函数, 并从标量表达式中调用此函数; ② 创建用户自定义过程, 并通过一个新的 SQL 语句 (典型的是 CALL) 调用这些过程。

继承性是 OODBMS 典型特性, M. B. Khaf<sup>[3]</sup> 提出四种借助关系表处理继承性的方法, 其核心是把分层结构中的每个类转换为一张表。

#### 2.1.2 数据操作转换

数据模式转换是指从 OODB 到 RDB 数据操纵语言 (DCL) 的转换。本文从 OODB 到 RDB 角度讲述。

数据库常用操作有数据查询、插入、删除和修改, 它们都离不开限制条件, 所以先讲述限制条件转换。

我们用  $Q_C$  和  $Q_R$  分别表示类限制条件和关系限制条件, 两者映射通常记作  $\rho_{Q_{UM}}(Q_C) \rightarrow Q_R$ 。相比之下,  $Q_C$  比  $Q_R$  多两个机制: 路径表达式操作数机制和集合操作数及运算符。通过直接设置类  $C$  某些属性及以类  $C$  为根类组合层次结构中的限定谓词得到  $Q_C$ 。常用限定谓词有原子谓词  $t_1 \theta t_2$  和量词  $\Theta(q^i c)$  (具体含义见文献 [4])。根据  $Q_C$  我们得到一个类限定图  $G_C$ 。同样每一个  $Q_R$  也对应一个关系限定图  $G_R$ 。有关  $G_C$  和  $G_R$  介绍见文献 [4]。

实施限制条件转换时, 通常是先根据  $Q_C$  构造  $G_C$ , 然后把  $G_C$  转换成  $G_R$ , 最后由  $G_R$  产生  $Q_R$ 。

数据查询转换是把对象查询运算转换为关系查询运算。其过程是从指定的类和 (或) 它的所有子类映射关系中选出于  $Q_C$  限定对象对应的元组 (由关系限定条件  $Q_R$  所限定)。

数据修改转换是把对象修改运算转换为关系修改运算。该操作受  $Q_R$  (由  $Q_C$  映射得到) 限定, 过程是删除所有旧元组后再插入新元组。数据插入转换与此相似。

数据删除转换是把对象删除运算转换为关系删除运算, 该操作受  $Q_R$  (由  $Q_C$  映射得到) 限定, 此时必须把与这些组合对象对应的元组中那些外关键字属性设置为 NULL。

### 2.2 模式演进技术

面向对象数据库中的类为适应需求变化而随时间变化称为模式演进, 包括创建新类、删除旧类、修改类属性和操作等。模式演进必须保持模式一致性 (模式自身内部不能出现矛盾), 这通过模式一致性约束来描述。模式一致性约束可分为唯一性约束 (同一模式中名字唯一)、存在性约束 (显示引用的成分须存在) 和子类型约束 (子类和父类的联系不可有环, 不能有从多继承带来的任何冲突等) 等, 满足所有这些一致性约束的模式称为一致模式。

模式演化历来是面向对象数据库研究的重点与难点。其解决途径一般有以下两种: ① 模式改变考虑现有应用程序, 使两者相互集成和适应。② 开发新的高级数据库编程语言。

常用演化方法有 TSE(透明模式演化<sup>[5]</sup>)、等价模式演化<sup>[6]</sup>和基于数据字典的模式演化<sup>[7]</sup>等。

## 2.3 索引技术

面向对象数据库数据庞大而复杂, 若无好的索引处理, 则数据处理效率十分低下。索引化过程就是对数据进行主体和特征分析, 赋予标志的过程。数据索引技术分为三种: 继承索引、集聚索引和集成索引。具体技术特点见文献[9]。

## 2.4 事务管理技术

### 2.4.1 传统数据库事务管理特点及不足

传统数据库事务管理由两部分组成: 并发控制和恢复。通常用锁协议实现并发控制, 用日志协议实现恢复。

在工程设计等应用领域需要长事务和协作事务。但事务原子性使得若长事务不能完成, 则所有已做的工作均要放弃; 而串行性则使得若长事务不结束, 其他事务必须等待, 造成传统并发控制方法效率极低。

### 2.4.2 OODBS 事务管理技术特点

OODBS 事务管理系统如图 2 所示。其中, 锁管理器管理锁表, 存放单个活动事务管理锁和等待锁。存储子系统与锁管理器实施对象上锁操作, 事务结束时释放此锁。死锁管理器检测和解除死锁。系统采用时间溢出技术, 即每个申请均有一时间限制, 时间溢出则死锁管理器将事务放弃。日志管理记录对象修改日志。

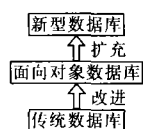


图 1 三种数据库之间关系

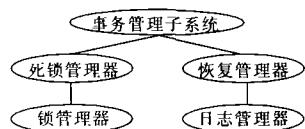


图 2 OODBS 事务管理系统

相比传统 RDBS OODBS 加锁技术的特点有: 加锁逻辑单位是对象而不是类; 给一个类对象加锁比给一个关系对象加锁需更多信息; 当一个类实例被加锁时, 其超类也被加锁。

数据库中被加锁项大小称为粒度。采用粗粒度锁机制时并发程序开锁代价低, 但系统并发行差; 采用细粒度锁机制则保证高度并发性, 但系统开锁代价大。

OODB 采用粗粒度加锁机制同样能达到很高的并行性, 加锁的一般是对象, 但是如果某一事务要访问同一个类的大多数实例, 则对整个类加锁, 既保证可靠性, 又降低系统开锁代价。

### 2.4.3 OODBS 恢复机制

RDBS 支持软故障和硬故障恢复。OODBS 采用来自软件故障和用户激活事务夭折的事务恢复, 不采用来自磁盘故障的恢复。OODBS 采用 UNDO 日志 (更新的对象页事务结束时存入磁盘)。恢复内容主要有以下两种:

(1) 多媒体日志恢复。通常 OODBS 把多媒体数据和其描述部分分开, 描述部分通过对象标识符引用相应多媒体数据。存储子系统管理一个能动态变化的空闲块链表, 里面存储多媒体数据, 系统把它作为多媒体数据日志。如果创建多媒体数据事务夭折, 则只需置空描述部分引用并将已分配给多媒体数据的空闲块链重新复位。同样如果删除多媒体数据事务夭折, 则

描述部分根据日志可恢复到原来状态。

(2) 索引页日志恢复。方法有两种: ① 分裂索引页的插入操作。系统把当前索引页的一半表项分给新页, 除了拷贝到新页的表项外, 其他表项都记录在日志中。发生故障时, 去掉整个新页即可。② 合并索引页的删除操作。系统把当前索引页的表项拷贝到新页, 除了从当前页删除的表项外, 其他表项都记录在日志中。发生故障时, 重新使用当前页即可。

### 2.4.4 虚拟事务

OODBS 支持两类事务: 常规事务和虚拟事务。前者提交时, 所有更新结果都永久地记录到数据库中; 它夭折时, 所有修改相当于没做。后者总是夭折的, 即不论此事务以什么方式结束, 对对象的改动不会记录到数据库中, 用户能通过此类事务对数据库进行复杂变化并观察结果, 不必担心不一致性。虚拟事务对象采用两个副本: 影子副本 (原始对象, 不被更新) 和当前副本 (被更新对象)。

### 2.4.5 长事务管理

传统事务模式处理长事务时存在冲突事务间长期等待和系统故障时数据库更新全部撤销两个缺点。OODB 中一个长事务可看作一些短事务集合。一个短事务看作并发控制和恢复的基本单位, 这样用户能减少锁粒度 (把长事务锁变成短事务锁), 实现不同长事务并发操作和长事务部分撤销。

## 2.5 视图类实现技术

### 2.5.1 面向对象数据库视图

传统数据库视图从某个特定角度反映数据库, 不存储数据, 也不占用空间, 但可当作实表操作, 也称为虚表。OODBS 中的视图具备传统数据库中的功能, 每个视图是一个“虚类”, 由一个或多个类产生, 虽不能产生对象实例, 但可当作对象实例操作。面向对象数据库中所有视图构成一个有向无环图, 其基本元素是对象视图类。对象视图类从模式中原类的某个查询推导产生, 它由属性和方法构成, 存在继承和合成关系。

### 2.5.2 面向对象数据库视图实现技术

面向对象数据库中很多操作 (如统计、连接查询和视图操作) 都能自由访问数据库数据, 利用这些操作实现 OODBS 视图操作, 能降低复杂度并提高效率, 但容易破坏对象封装性。

为了不破坏对象封装性, 我们在对象中设计一组接口, 系统通过这组接口完成视图操作, 这样会增加对象复杂性和 OODBS 设计难度。为了克服这个缺点, 我们对这些接口实行标准化, 把它们与数据库中其他对象的服务结合。

基于上述条件, 我们设计相应类数据结构和操作实现 OODBS 视图。具体算法见文献[12]。

### 2.5.3 面向对象数据库视图集成技术

视图类定义好后, 我们把它们集成在一起构成有向无环图, 其基本元素是对象视图类。视图类集成过程算法见文献[13]。

## 2.6 版本管理技术

工程类应用中设计工作随时间逐渐演进, 本身就是一个不断反复、试探、选择和完善的过程, 其间会产生同一被设计对象的多个版本, 它们必须妥善管理。为了降低设计复杂性, 常常采用分层逐步细化的方法。这样, 一个被设计对象由多个子对象构成, 每一个子对象同样产生多个版本。子对象某些版本合

起来就构成了上层对象某个特定版本,并且如果某个子对象创建一个新版本,上层对象可能派生一个对应的新版本,等等。此外,在模式演化过程中,常用版本管理控制对象演化过程。

版本管理有两个方面:①集合管理。对所有版本管理,其关系有两种,即时间先后关系,是最基本的关系,一般用版本号表示,以及派生关系,如图3所示,这种图叫做版本图。版本集合管理常用版本图进行管理。②引用管理。多版本系统中的对象只是逻辑上虚拟的概念,实际存在的是该对象的各个版本,所以,使用对象就是引用它的某一版本。一般有两种引用方法:静态引用(直接引用某个对象的特定版本)和动态引用(引用关系指向某个对象,不一定是哪个版本)。相比之下,动态引用更有效,更贴近实际。

根据版本是否已“冻结”,版本分为发行版本(已经定型和“冻结”,不可更改)和临时版本(可修改)两种。

## 2.7 安全建模技术

随着 Internet 技术不断发展,安全性已成为不可忽视的问题,利用安全模型能精确地描述系统安全策略。安全模型和数据库系统的结合就是数据库安全建模技术。

OODB 通常结合 RDB 安全技术描述 OODB 安全技术。常用 OODB 安全模型有支持单级和多级对象的两种模型。相比之下,多级对象模型能更好地描述现实世界实体,是实现 OODB 安全性的主流。

### 2.7.1 安全建模基本框架

安全建模本质是利用面向对象建模技术,对现实世界各种安全性引入若干种安全性约束分类,进行安全性分等,将现实系统中的安全性语义表达成数据库系统支持的安全性模型。在此过程中可能会产生冲突,引起数据库安全性语义的不一致性,因此,进行一致性检测和解决冲突是必要的。整个框架如图4所示。

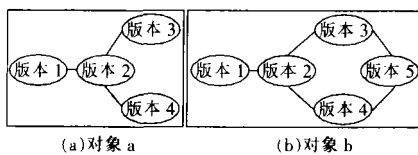


图3 版本图

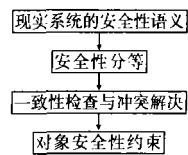


图4 安全建模过程

由图4可知,安全建模主要有两个任务:安全性分等和一致性检查与冲突解决。其中的一致性检查与冲突解决任务由机器完成。安全性分等是由 OODB 提供方法,由应用系统设计者(建模者)完成。相比之下,安全分等更为重要,对此我们着重介绍。

### 2.7.2 基本概念

定义1 对象类。一个对象类为  $O(A_1, A_2, \dots, A_n, M_1, M_2, \dots, M_m)$ , 其中  $A_i (i=1, 2, \dots, n)$  为对象属性, 定义在域  $D_j$  上;  $M_j (j=1, 2, \dots, m)$  为对象方法, 返回值在域  $D_j$  上。引入多级对象安全模型后, 对象类被扩展为多级对象类, 表示为  $O^m(A_1, C_1, \dots, A_n, C_n, M_1, C_{n+1}, \dots, M_m, C_{n+m}, CO)$ , 其中  $C_i (i=1, 2, \dots, n)$  为属性  $A_i (i=1, 2, \dots, n)$  对应安全标志,  $C_{n+j} (j=1, 2, \dots, m)$  为方法  $M_j (j=1, 2, \dots, m)$  对应安全标志,  $CO$  是整个类安全标志。

定义2 关联。对象间关系叫做关联。按照参加对象数目分成一对一、一对多和多对多三种。

定义3 聚集是一种组合关系, 指一个对象能由其他对象

组成, 体现 part of 关系。

定义4 特化。特化是一种继承关系, 子类能继承父类的属性和方法, 也能定义自己的属性和方法, 体现 is a 关系。

### 2.7.3 安全分等

现实对象  $O$  加上适当安全约束, 成为多级对象  $O^m$  的过程叫做安全分等。

应用时, 不仅对象属性和方法要安全分等, 对象之间的关联、聚集、特化、概括关系和 OODB 的查询结果也要安全分等。安全分等有三种: ①对属性分等。设定属性值安全级别。②对方法分等。对方法本身标志安全级别。③对对象实例分等。确定某个对象实例的属性和方法的安全级别范围。

### 2.7.4 安全分等实现

安全分等实现过程就是根据应用系统的安全性语义, 将各种安全性约束附加于对象。

安全性约束有单体和集合约束两种。单体约束对单个对象进行安全性约束, 安全程度低, 有简单约束、谓词约束、复合约束和级别约束四种; 集合约束对查询结果集合进行安全性约束, 安全程度高, 有聚集约束和推理约束两种。

## 3 面向对象数据库的几种实现方法

面向对象数据库实现途径一般有以下两种:

(1) 以一种面向对象语言为基础, 增加数据库功能, 建立纯粹的面向对象数据库管理系统 (OODBMS), 主要目标是实现数据共享和支持持久对象。这种数据库不仅在处理多媒体等复杂数据类型时游刃有余, 而且在应用系统开发速度和维护等方面有着极大的优越性。但是, 它们不支持 SQL 语言, 在通用性以及其他软件兼容等方面失去优势, 因而其应用领域有很大局限。典型代表有 Servio 公司开发的 GEMSTONE Ontologic 公司开发的 ONTOS 等。

(2) 对传统的关系数据库系统加以扩展, 增加面向对象特性, 把面向对象技术和关系数据库相结合, 建立对象关系数据库管理系统 (ORDBMS)。这种系统既支持已经被广泛使用的 SQL, 具有良好的通用性, 又具有面向对象特性, 支持复杂对象和复杂对象的复杂行为, 是对象技术和传统关系数据库技术的最佳融合。典型代表有 Oracle 8 等。

## 4 面向对象数据库的发展前景

面向对象数据库是面向对象技术和数据库技术结合的产物, 利用它能处理海量和复杂数据的特征, 可以弥补传统数据库系统诸多不足, 满足蓬勃兴起的网络技术、多媒体技术、空间信息科学、信息管理、人工智能、软件工程技术和数据挖掘技术等诸多学科的发展需要。OODBS 是新型数据库的基础, 与上述学科结合能形成专用数据库用于学科研究。在商业领域和工业设计领域, 尤其在处理长事务和并发控制方面, OODBS 大有作为, 独树一帜。尽管 OODBS 诞生时间不长, 还有很多缺陷, 但随着研究和开发, OODBS 会变得更加完善。从笔者的角度来看, 理论上 OODBS 能借助 RDBS 成熟研究理论完善自身, 形成较完善的理论体系; 应用上积极结合其他学科, 获得更大的发展空间。这样, OODBS 就会获得更快发展, 最终成为 DBS 主流。

(下转第 34 页)

证,根据“公理:所有 $X$ :中医舌苔,出现概率大于( $X$ :主证, $X$ :次证)”可以推出出现黄苔时是里证和热证的概率是大于虚证的,这是符合临床情况的。

## 6 总结

本文探讨了中医舌诊知识获取和建立舌诊本体的方法,我们已经建立了一个比较完备的中医学本体体系,中医舌诊部分是对应的完备和补充。中医学是我们的国粹,在拯救人民于疾苦的过程中扮演着重要的角色。现在我国中医领域正在积极探索促进中医信息化、现代化方法,我们也想为中医学走向世界做点工作。我们现在建立的 NKI 中医学知识库有着广阔的应用前景,就拿本文阐述的中医舌诊知识来说,它可以为辅助中医专家诊断、中医专家系统、中医智能教学系统等提供所需要的知识,因此,它的重要性不言而喻。但是中医知识的获取却不是一件很容易的事情,这不但要求知识工程师对中医知识有很好的理解,同时又要求有很扎实的计算机科学知识。因此在以后的工作中,我们将不断探索中医知识获取好的方法,发现知识获取的技巧,这些工作有着十分重要的意义。

### 参考文献:

- [1] Natalya F Noy, Mark A Musen. SMART: Automated Support for Ontology Merging and Alignment [C]. Banff Canada. Proceedings of the 12th Workshop on Knowledge Acquisition, Modeling and Management, 1999.
- [2] Y Sun, S Staab, J Angels, et al. OntoEdit Guiding Ontology Development by Methodology and Inferencing [C]. Lyon France. Prestigious Applications of Intelligence System (PAIS), 2002.
- [3] 陆汝铃. 世纪之交的知识工程与知识科学 [M]. 北京:清华大学出版社, 2001.

(上接第 4 页)

## 5 结束语

本文作为一篇综述性文章,总结了 OODB 常用最新技术原理和方法,叙述了 OODB 两种常用实现技术,并且对 OODB 的发展前景作了预测。OODB 作为一项崭新技术,具有前两代数据库无法比拟的优点,满足复杂数据结构和海量存储需要,在各个学科大有发展之地,最终将会成为 DBS 的主流。

### 参考文献:

- [1] 王春森. 系统设计师(高级程序员)教程 [M]. 北京:清华大学出版社, 2001.
- [2] 钟珞. 系统分析员重点综述与试题分析(第 4 版) [M]. 北京:中国民航出版社, 2003.
- [3] M Blaha, W Pnmerkni, H She. Converting OO Models into RDBMS Schema [J]. IEEE Software, 1994, 11(1): 28-39.
- [4] 王学荣,曾晓勤. 面向对象数据库到关系数据库运算的转换 [J]. 计算机应用与软件, 2003.
- [5] Gook R Y. A Transparent Schema evolution System Based on Object Oriented View Technology [J]. IEEE Transactions on Knowledge and Data Engineering, 1997, 9(4).
- [6] 董传良,陆嘉恒,董玮文,等. 面向对象数据库中的等价模式演化策略 [J]. 计算机工程, 2000, 26(9): 23-24.
- [7] 张民强. 利用数据字典实现面向对象数据库的模式演进 [J]. 应

- [4] 曹存根. 面向专家的知识获取 [M]. 北京:科学出版社, 1998.
- [5] 曾庆田. 数学概念的知识获取与分析 [D]. 北京:中国科学院计算技术研究所, 2005.
- [6] Cungen Cao. Extracting and Sharing Medical Knowledge [J]. Journal of Computer Science & Technology, 2002, 17(3).
- [7] 辛瑛,郭霞珍,张莉莎. 舌诊 [M]. 天津:天津科技翻译出版公司, 2001.
- [8] 张德海,曹存根,张宇翔. 国家和城市知识获取与本体论分析 [C]. 中国人工智能学会.
- [9] 唐素勤,曹存根. 智能教学系统:综述与改进 [C]. 中国人工智能学会第九届全国学术年会暨中国人工智能学会成立 20 周年庆祝大会, 2001, 1129-1132.
- [10] 《中国大百科全书》之中医卷 [M]. 北京:中国大百科全书出版社.
- [11] Cao Cungen, Feng Qiang, Gao Ying, et al. Progress in the Development of National Knowledge Infrastructure [J]. Journal of Computer Science & Technology, 2002, 17: 523-534.
- [12] Cungen Cao. Technology Focus of 21st Century [N]. Computer World, 1998(D1-D3).
- [13] Cungen Cao. Medical Knowledge Acquisition from the Electronic Encyclopedia of China [C]. Lecture Notes in Computer Science, 2001, 268-271.
- [14] Cungen Cao, et al. Progress in the Development of National Knowledge Infrastructure [J]. Journal of Computer Science & Technology, 2002, 17(5): 1-16.
- [15] C Cao, H Wang, Y Sui. Knowledge Modeling and Acquisition of Traditional Chinese Herbal Drugs and Formulae from Text [J]. International Journal of Artificial Intelligence in Medicine, 2004, 33: 3-13.

### 作者简介:

曹宇峰,硕士研究生,主要研究方向为人工智能、知识获取、智能教学;曹存根,研究员,博士生导师,主要研究方向为人工智能、知识工程、大规模知识处理、情感计算。

用科技, 2003, 30(10): 24-26.

- [8] 程胜,湛潜,董金祥. 面向对象数据库中的索引实现技术 [J]. 计算机工程, 1999, 25.
- [9] 温有奎. 面向对象数据库系统标引 [J]. 情报杂志, 2000, (1): 50-51.
- [10] 陈仲民,王雪. 面向对象数据库的索引技术 [J]. 计算机工程与科学, 2000.
- [11] 郑刚. 面向对象数据库的事务管理 [J]. 微机发展, 2002.
- [12] 王兰生,尹湛,居梯. 面向对象数据库视图的研究和实现 [J]. 南京邮电学院学报(自然科学版), 2000.
- [13] 马新民,王建. 面向对象数据库视图类推导及集成算法 [J]. 安徽大学学报(自然科学版), 2004.
- [14] 徐立臻,徐宏炳,陈刚. 面向对象数据库系统中的版本管理 [J]. 东南大学学报, 1999.
- [15] 罗慧. 面向对象数据库中基于版本管理的模式演进方法的改进 [J]. 微型机与应用, 2003, 22(12): 6-8.
- [16] 王赫然. 面向对象数据库的发展与现状 [J]. 铁路计算机应用, 1999.
- [17] 龙海泉,汤彬. Oracle 面向对象数据库的实现 [J]. 南华大学学报(理工版), 2003.

### 作者简介:

王功明(1981-),男,河南开封人,硕士研究生,主要研究方向为智能信息技术;关永(1966-),男,辽宁凤城人,副教授,硕士生导师,博士,主要研究方向为智能信息处理系统;赵春江(1964-),男,河北定县人,研究员,博士生导师,博士,主要研究方向为农业信息化及智能化应用;王蕊(1984-),女,北京密云人,本科生,专业方向为信息工程。