第33卷 第10期 计 算 机 工 程 2007年5月
Vol.33 No.10 Computer Engineering May 2007
·软件技术与数据库· 文章编号：1000—3428(2007)10—0076—03 文献标识码：A 中图分类号：TP 311.131

基于元目录的数据管理模型的研究与实现

王建芳[1,2]，阎保平[1]，吴开超[1]，沈志宏[1]
（1. 中国科学院计算机网络信息中心数据库应用研究室，北京 100080；2. 中国科学院研究生院，北京 100080）

摘 要：提出了一种新的基于元目录的数据管理模型，给出了一个应用案例。该模型结构灵活、适应性强，可以有效实现数据整合、数据实时共享和数据可控，适用于大规模数据的数据整合和数据应用。
关键词：元目录；存储模式；基模式；结构模式；表现模式

Research and Implementation of Meta-directory-based Data Management Model

WANG Jianfang[1,2], YAN Baoping[1], WU Kaichao[1], SHEN Zhihong[1]
(1. Scientific Database Department, Computer Network Information Center, Chinese Academy of Sciences, Beijing 100080;
2. Graduate School, Chinese Academy of Sciences, Beijing 100080

【Abstract】 This paper puts forward a new data management model: data management model based on meta-directory. An application instance is given to make the model understood more clearly. This new model has many advantages such as flexibility and adaptability. The model is applied in many practical applications such as integrating data, on-time sharing data and controlling data. It is especially useful in integrating and applying abundant data.
【Key words】 Meta-directory; Storage schema; Base schema; Structure schema; Presentation schema

传统的关系型数据管理系统结构简单、建库方便而且编程接口灵活，因而得到了广泛的应用。但在应用的过程中也暴露了许多缺点，环境适应性差是其中一个很明显的缺陷，主要表现在以下几点：在要求系统频繁改变的环境下，关系型数据管理系统的成本高且修改困难；在工程应用中支持“模式演变”（Schema Evolution）的功能很重要，而RDBMS不易支持这种功能；可移植性差，从一个环境转换到另一个环境时需要多至30%的附加代码[3]。在当今的信息时代，数据繁多而复杂，数据更新又极快，导致这些缺点更加明显，这将制约信息管理的发展。如何提高数据管理的环境适应性成为亟待解决的问题。

1 元目录技术

元目录（meta-directory）技术即元数据目录技术，它将分布的数据资源组织管理起来，为用户查询提供服务，关注的是数据集而非数据本身。本文对数据集的定义是描述基本数据集的信息。

从结构上看，元目录提供的信息包括：数据集的内容，数据集的位置，数据集的访问方式，数据集的访问权限，以及数据集的组织方式等信息。

元目录技术有很多优点：可以将分散的数据整合成一个单独的目录，将先前相互隔离的信息集成到一个视图中，提供一个集成化的一致视图；实现数据同步；控制数据的更新方式。可提供通用的访问模式，能通过多种数据源命名、搜索或升级数据。元目录的灵活性使它适应任何企业的组织、结构、策略和管理风格；并且有足够的活力随着它们的变化而变化。

目前，元目录主要应用于对分布式的资源进行整合和统一管理。它还可以应用于数据管理模型中，设计出一种新的管理方便、应用灵活的数据管理模型。这种基于元目录的数据管理模型建立在关系型数据管理模型的基础上，定义了4层模式的体系结构，使它不仅拥有关系型数据管理模型的优点，而且还具有通用性、灵活性和很强的环境适应性，这些是关系数据模型无可比拟的。

2 基于元目录的数据管理模型

2.1 基本思想

传统的关系模型将事物间的关系看作关系表来管理数据，一旦模式改变，用户就不得不更改应用程序，使其适应。本模型的基本思想是：(1)定义字段，这里称作维；(2)用户可以从已定义的维中选择某一子集作为结构表模式；(3)用户导入数据源后形成事实表；(4)应用程序可以根据应用需求访问事实表的记录子集，形成视图。如图1所示。

图1 模型的基本思想

这样的设计使数据定义更加灵活，数据更新更加方便，大大提高了环境适应性。

2.2 模式结构

该模型采用4层模式的体系结构，由下至上分别为存储

基金项目：中国科学院“十五”信息化建设基金资助重大项目（INF105-SDB）
作者简介：王建芳(1981－)，女，硕士生，主研方向：计算机软件理论；阎保平，博士、研究员、博导；吴开超，高工；沈志宏，工程师
收稿日期：2006-07-10 **E-mail：**jfwang04@sdb.cnic.cn

模式、基模式、结构模式和表示模式。模式结构如图 2 所示。

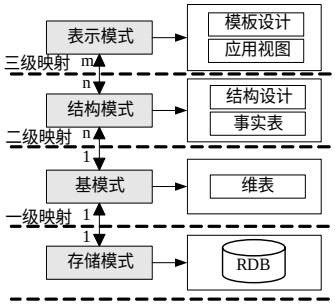

图 2 四层模式结构

下面介绍本模型在各层模式是如何设计的，以及如何通过三级模式映射实现各层模式之间的独立性和灵活性。

2.2.1 存储模式

存储模式（Storage Schema），又称平台模式（Platform Schema），它是数据物理结构和存储方式的描述。

由于在分析比较现行各种数据库后发现，关系数据库具有很多优点（如结构简单、使用方便、应用广泛等），而且它的思想与本模型的思想相吻合，因此模型采用关系数据库作为平台，由此保证模型的易设计性、易用性和兼容性。

2.2.2 基模式

基模式（Base Schema），也称维模式（Dimension Schema），由一些基础的维组成。这些维是由描述核心数据相关的维度衍生的，包括基础维 BD（Basic Dimension）和扩展维 ED（Extension Dimension）2 种。基础维有行维和列维。扩展维是由用户定义的满足一定规范的维，既有结构性，又有一定的灵活扩展性。

假设：B 是基础维，E 是扩展维，D 是基模式中的所有维，则存在如下约定：

B={列维}或 B={行维，列维}；|E|∈N；D=B∪E。

2.2.3 结构模式

结构模式（Structure Schema）包括结构设计和事实表 2 部分。

结构设计是用户根据需求，从基模式中的维中选择某一子集构成一定的结构表。结构表集S的定义：S⊆B'×E'，其中，B'⊂B且E'⊂E。

事实表是用户按照某种结构将数据源重构形成的新表。

结构表的设计定义了事实表的模式，事实表和相关结构表中的维之间的关系采用星状模型设计，如图 3 所示。

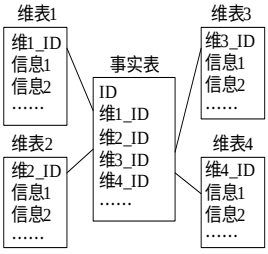

图 3 星状模型

2.2.4 表示模式

表示模式（Presentation Schema）面向具体的应用程序，是用户与数据库系统的接口，是用户用到的那部分数据的描述。它包括模板设计和应用视图 2 部分。二者都是关于用户应用的设计，但模板设计侧重于结构设计，如选择哪些维对应的数据进行显示。而应用视图侧重于表现风格设计，如数据按数据表还是 XML 的风格显示等。这样的设计使机构和表现风格相分离，增加了模型的灵活性和适应性。

2.2.5 模式映射

该模型中有三级映射，使得各个层次之间实现松散耦合，保证了层之间的独立性和结构的灵活性。

为了便于描述映射规则，首先定义一些符号。

存储模式中的关系表 T 可用三元组表示为<TNAME, TID, TC>。其中，TNAME 表示表 T 的名称；TID 表示表 T 的主键；TC 表示表 T 的所有列（主键除外）的集合。

基模式中的维 D 可用三元组表示为：<DNAME, DID, DP>。其中，DNAME 表示维 D 的名称；DID 表示维 D 的标识符属性；DP 表示维 D 的其他属性的集合。

结构模式中的结构 S 可用四元组表示为<SNAME, SID, DIDS>。其中，SNAME 表示结构 S 的名称；SID 表示结构 S 的标识符；DIDS 表示一些维 ID 的集合。

表现模式中的模板 M 可用四元组表示：<MNAME, MID, DIDS>。其中，MNAME 表示模板 M 的名称；MID 表示模板 M 的标识符；DIDS 表示一些维 ID 的集合。

(1)一级映射：存储模式和基模式之间的映射。存储模式中的关系表和基模式中的维存在一一对应的关系。一个维 D<DNAME, DID, DP>到一个关系表 T<TNAME, TID, TC>的映射为 f，f 满足如下条件：DNAME=TNAME；DID=TID；对∀p∈DP，∃c∈TC 与之相对应。

(2)二级映射：基模式和结构模式之间的映射。基模式与结构模式存在 1 对多的关系，即在一个基模式上可以构建多个结构模式。一个结构 S<SNAME, SID, DIDS>到维集 DS＝{x 是一个维度|x=<DNAME, DID, DP >}的映射为 f，f 满足如下条件：对∀did∈DIDS，∃d∈DS，且 d.DID=did。

(3)三级映射：结构模式和表现模式的映射。结构模式与表现模式存在多对多的关系。一个模板 M<MNAME, MID, DIDS>到一个结构 S<SNAME, SID, DIDS>的映射为 f，f 满足如下条件：对于 M.DIDS⊆S.DIDS。

2.3 元目录管理

元目录技术为模型提供了元目录的管理方案。它把各层的数据资源都整合成一个目录，便于统一管理。

元目录管理分为 2 级：系统级管理和应用级管理。系统级管理关注数据集（数据库）本身的结构，而应用级管理与应用有关，主要关注数据库的显示度，它构建在系统级元目录的基础上。系统级元目录与应用级元目录在逻辑上是分开的。元目录采用统一的元数据存储方式，即系统中所有的元数据都存放在基于 LDAP 的目录中，进行统一管理。元目录管理如图 4 所示。

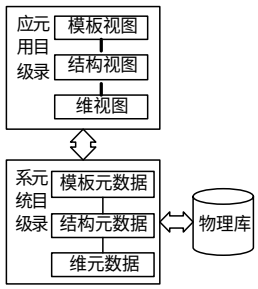

图 4 元目录管理

3 实例应用

目前，本模型已经成功应用于中国科学院的重点项目“科学数据库及其应用系统”，开发实现了一个子系统“中国能源数据库通用数值型数据管理系统”。该系统主要对能源领域的

数值型数据进行管理。本文将其作为一个应用实例来验证模型的可行性。

3.1 实例框架

模型可以有各种实现框架，本系统采用了基于中间件技术的实现框架。如图 5 所示。

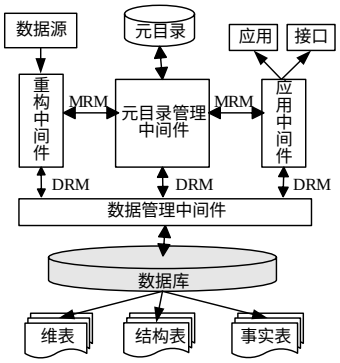

图 5 实例框架

在本框架中，功能的实现基于 4 个中间件：重构中间件负责将数据源重构到关系数据库中，包括向用户提供一个方便的 GUI，进行维管理、结构表管理、事实表管理等；应用中间件为用户提供应用服务功能，同时为编程人员提供扩展应用的接口；元目录管理中间件则负责管理元目录，并为其他部件提供元目录信息；数据管理中间件实现其他部件对数据的操作。由于客户多与重构中间件和应用中间件交互，因此将二者统称为客户端中间件。通过该框架，用户可以方便地将数据信息重构到模型中，对数据信息进行有效的管理和应用。

3.2 实例流程

在该系统中，具体的管理流程如下：

(1)用户通过客户端中间件提出应用请求，并输入一些必需的参数（Parameter）；

(2)客户端中间件分析客户请求和参数（Parameter），生成一个元目录查询请求，以元目录请求消息(Meta-directory Request Message，MRM)的方式传递给元目录管理中间件；

(3)元目录管理中间件根据请求查询所请求的信息，并将查询结果和请求消息一起封装成一个数据请求消息（Data Request Message，DRM），发给数据管理中间件；

(4)数据管理中间件接到请求，到相应的库对相应的数据表项进行数据操作；

(5)最后将操作结果反馈给用户。

3.3 访问控制策略

良好的访问控制策略对于保证信息被安全合理地利用具有重要的意义。在模型的实现方案中，必须制定合适的访问控制策略，这样不仅保证了数据资源被合法访问和合理使用，而且可以使各个部件各施其职，充分发挥系统的优势。

现存的访问控制策略有多种，本系统采用入网权限控制和基于角色控制相结合的方法。首先由入网权限控制确定哪些用户能够登录到服务器并获取网络资源，以及用户入网时间和入网地点，并为其分配角色。然后，角色访问控制根据用户角色指派的权限来控制用户对目录、文件、设备的访问。

角色权限控制通过元目录技术实现，将目录和文件看作结点进行管理，从而通过控制结点控制相应的数据信息。

角色对目录和文件的访问权限一般有 8 种：系统管理员（Supervisor）权限，读（Read）权限，写（Write）权限，创建（Create）权限，删除（Erase）权限，修改（Modify）权限，文件查找（File Scan）权限，存取控制（Access Control）权限。用户在目录一级指定的权限对所有文件和子目录有效，用户还可进一步指定对目录下的子目录和文件的权限。

4 结束语

本文提出了一种结构灵活、适应性强的数据管理模型——基于元目录的数据管理模型。通过该模型，用户可以将数据整合到一个统一的平台，管理方便灵活，在中科院信息建设中起到了重要的作用。将来，可考虑将其应用于大规模数据的存储和管理，以及在此基础之上的应用开发，为海量信息挖掘提供了一个很好的数据模型基础。

参考文献

1 Tillery S. The Metadirectory: a Directory for the Real World[J]. Messaging Magazine, 1999, (7/8).

2 Sun Microsystems Inc.. The Sun one Meta-directory Deployment Guide[R]. http://docs.sun.com/source/817-3898-10/preface.html.

3 莎师煊, 王　珊. 数据库系统概论[M]. 3 版. 北京: 高等教育出版社, 2000.

4 于　丹. 元目录：企业目录新走向[Z]. http://www.pcworld.com.cn/99/9919/1931a.asp.

5 沈志宏, 王龙潇. 目录型元数据在科学数据库系统平台中的应用[Z]. 中国科学院计算机网络信息中心科学数据库中心, 2004.

(上接第 75 页)

4 结束语

本文介绍了一种利用模糊数学手段构建基于语义数据仓库的方法，从数据加工的角度出发，阐述了数据加工的 3 层结构，同时，为了实现快速的模糊查询，完成对模糊数据的有效管理，提出一种在数据仓库内部建立模糊数据管理模块的方法，在不改变查询语言的基础上，通过该模块中的模糊数据元数据实现对模糊数据的管理和利用。

参考文献

1 Immon W H. Building the Data Warehouse[M]. 王志海, 林友芳, 译. 北京: 机械工业出版社, 2003.

2 Ling Feng, Dillon T S. Using Fuzzy Linguistic Representations to Provide Explanatory Semantics for Data Warehouses[J]. IEEE Transactions on Knowledge and Data Engineering, 2003, 15(1): 86-102.

3 Alhajj R, Kaya M. Integrating Fuzziness into OLAP for Multidimensional Fuzzy Association Rules Mining[C]//Proc. of the 3th IEEE International Conference on Data Mining. 2003.

4 Au W H, Chan K C C. Classification with Degree of Membership: A Fuzzy Approach[C]//Proceedings of IEEE International Conference on Data Mining. 2001.

5 尤玉林, 张宪民. 一种可靠的数据仓库中的 ETL 策略与构架设计[J]. 计算机工程与应用, 2005, 41(10): 172-174.