第 35 卷 第 9 期 计 算 机 工 程 2009 年 5 月
Vol.35 No.9 Computer Engineering May 2009
·软件技术与数据库· 文章编号：1000—3428(2009)09—0079—03 文献标识码：A 中图分类号：TP391

基于银行数据仓库的元数据管理系统

谢福成，王备战，史 亮，姜青山
(厦门大学软件学院，厦门 361005)

摘 要：元数据在银行数据仓库中具有重要意义。讨论元数据的分类和作用，分析元数据的管理功能，给出一种基于银行数据仓库的元数据管理系统，对其结构进行分析和说明。实践结果表明，该系统可以加强对数据的分析和管理能力，提高银行数据仓库等决策系统的灵活性和可扩展性。

关键词：元数据；元数据管理；数据仓库

Metadata Management System Based on Bank Data Warehouse

XIE Fu-cheng, WANG Bei-zhan, SHI Liang, JIANG Qing-shan
(Software School, Xiamen University, Xiamen 361005)

【Abstract】Metadata plays an important role in bank data warehouse. This paper discusses the categories and functions of metadata, and analyzes the management function of metadata. It presents a metadata manage system based on bank data warehouse, analyzes and illustrates its structure. Application results show that this system can increase the ability of data analysis and management, and improve the flexibility and expansibility of the decision system such as bank data warehouse.

【Key words】metadata; metadata management; data warehouse

1 概述

随着金融国际化速度的加快，各商业银行间的竞争日趋激烈。各大银行建立了自己的数据仓库应用平台，用于加强经营管理和决策支持，并更好地了解客户需求，从而开发新产品或服务，以提高竞争力。数据仓库平台能对大量业务信息进行快速的综合处理与分析，提升业务运作效率和客户服务水平，增加赢利能力，并在特定业务领域提供差异化服务。银行数据仓库的元数据是银行数据仓库实现的基础，它规范了银行数据仓库中的数据来源、数据抽取和转换规则以及目标数据模式等，有利于数据仓库数据的管理、使用和共享。通过对银行数据仓库元数据进行密集型集成和管理，可以建立真正支持数据挖掘分析处理的银行数据仓库。可见，元数据管理对于银行数据仓库而言具有重要意义[1]。本文研究银行数据仓库元数据技术，从系统开发实践角度出发，结合具体项目开发经验，设计并实现一个基于银行数据仓库的元数据管理系统。

2 银行数据仓库元数据

2.1 银行数据仓库元数据的概念

元数据是关于数据的数据，即描述流程、信息和对象的数据。银行数据仓库元数据是关于银行业务数据和技术数据的数据，用来描述银行数据仓库中的主题信息、外部数据源和非结构化信息、物理和逻辑数据模型、数据的抽取和转换规则、数据的粒度和分割定义、数据和质量的管理方式以及其他相关业务数据信息。它主要包括建立银行数据仓库过程中的数据需求、模型设计和 ETL 操作等环节产生的文件类数据(如 Excel 文件)、数据类数据(如数据模型)、业务主题类数据、IT 操作类数据、数据接口类数据、业务过程指标类数据和其他类数据[2]。

2.2 银行数据仓库元数据的分类

由于不同领域、不同行业对元数据的研究角度不同，因此其分类相同，例如，按抽象层次分类可以分为概念元数据、逻辑元数据和物理元数据；按来源分类可以分为工具元数据、资源元数据和外来元数据。在银行数据仓库环境下，根据元数据用途及其针对使用角色的不同，可以分为技术元数据和业务元数据。技术元数据面向技术开发人员，是技术开发、系统维护和改进的基础，主要包括文件类元数据、数据类元数据(主要指物理模型)、IT 操作类元数据和数据接口类元数据 4 个部分。业务元数据面向业务分析人员，是对数据和处理规则的业务化描述，主要包括数据类元数据(主要指逻辑模型)、过程和指标类元数据、报表类元数据和业务主题管理类元数据 4 个部分。

2.3 银行数据仓库元数据的作用

银行数据仓库元数据能有效帮助技术人员和业务人员理解、监督和管理数据来源、业务主题以及转换规则、数据变更和单元调度等信息，从而提高开发工作效率，保证银行数据仓库能高效准确地建立和运作，其作用可以归纳为以下 4 个方面[3]：

(1)集中式的元数据管理模式能有效提高技术开发人员和数据分析人员对数据库开发和数据分析使用的效率。

(2)提供良好的元数据查询管理应用界面，可以使业务人

基金项目：国家自然科学基金资助项目“高维数据聚类的数学模型及其在反垃圾邮件中的应用”(10771176)

作者简介：谢福成(1983－)，男，硕士，主研方向：数据仓库，元数据；王备战，教授、博士；史 亮，副教授、博士；姜青山，教授、博士

收稿日期：2008-11-18 **E-mail：**mianbao1983@gmail.com

员独立准确地定位和使用数据仓库中的有效信息。

(3)提出元数据管理组织、元数据管理的标准和流程，准确定义元数据范围，设置相关管理工作的人员，进而保证商业银行数据仓库系统中元数据的完整性和正确性。

(4)能有效支持商业银行 IT 系统的维护和需求改进，促进不同 IT 系统间的数据交换。

3 银行数据仓库元数据系统的设计

3.1 元数据管理系统的体系结构

元数据系统管理平台的建立使技术人员、业务人员和其他相关人员可以统一对银行数据仓库系统中各类元数据进行管理和监督。本文结合 CWM(Common Warehouse Metamodel)规范[4]，采用 3 层 J2EE 体系架构设计该系统，全部程序可以无缝地在不同应用平台间进行移植，整体分为 3 个部分：桥接器，元数据存储和前端界面。桥接器通过加载程序和 MDS(Meta Data System)知识库进行会话。界面通过 HTTP 方式与数据库进行连接。元数据管理系统的体系结构如图 1 所示。

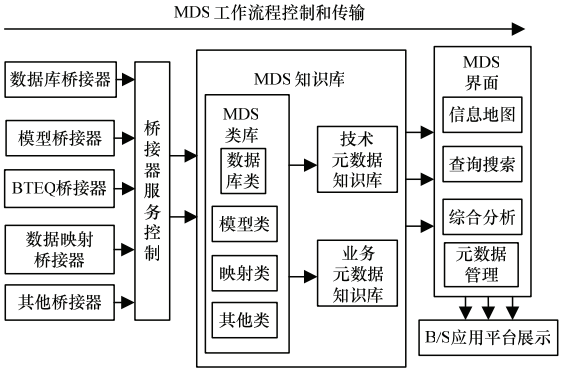

图 1 元数据管理系统的体系结构

元数据管理系统体系结构描述如下：

(1)桥接器(适配器)。又称元数据采集部分，其主要功能是使用不同桥接器对数据库类、映射类、模型类等元数据进行抽取、转换并加载到 MDS 知识库中。

数据库桥接器用来从不同数据库中读取指定数据库表的字典信息，并将其转换成相应的元数据信息，存放在元数据库中。字典信息包括每个物理表的详细信息，如数据库名称、表名称、字段名称、字段类型、字段说明、是否主键、是否为空等特征。它通过页面触发的形式，由系统自动访问数据仓库系统，读取数据库类元数据，并加载到元数据管理系统的数据库。与之前版本的元数据进行自动比对，生成元数据差异统计报告，供元数据管理员检查核对并发布。

BTEQ 桥接器用于解析各种用来处理数据转换规则的脚本，如 Teradata 数据转换脚本等。解析器能从脚本中抽取各种数据转换关系，包括转换算法、转换路径等，并存入元数据库相关信息表中。此桥接器使用 JavaCC 语法解析器作为核心解析框架，通过替换 SQL 语法定义，能支持解析多种数据库厂商 ETL 转换工具对应的脚本。在解析过程中，脚本解析器会把不符合规范的脚本写入日志。

模型桥接器主要完成从模型工具生成的源文件，例如，从 ERWin 工具导出的 XML 文件中解析出模型实体和关联关系信息，并转换成元数据库中的元数据信息。在解析模型文件前，解析器会对文件有效性进行统一检查，不符合检查规则的模型文件将在日志中给出报告。

数据映射桥接器主要负责把人工编写的数据映射文档导入到元数据存储库，或从数据仓库生产环境的 ETL 脚本抽取数据映射关系到元数据存储库。该类元数据信息主要包括类元数据，如任务名、任务描述、目标表描述、目标字段赋值描述、源表描述等信息。

其他桥接器主要包括 BI 工具桥接器和 XML 工具桥接器。BI 工具桥接器用来从不同数据库中读取指定数据库表的字典信息，并转换成相应元数据信息，存放在元数据库中。XML 桥接器用来解析符合一定模版的 XML 文件。

(2)MDS 知识库存储部分。设计合理的元数据存储结构，整合并存储来自各种渠道的元数据信息，如数据库类元数据、模型类元数据等。汇总到存储部分的元数据信息首先会被分成技术和业务 2 大类信息，然后依据不同使用主题进行组织和细分，以提高后期元数据查询和分析等应用的效率。

(3)数据应用部分。实现信息地图、查询搜索、变更管理、综合分析、血缘分析和影响性分析等功能应用。元数据变更是对有变化的元数据进行修订，对新的元数据进行采集或抽取，即完成各类元数据的增加、修改、禁用，具体可以分为个别变更和批量变更 2 类。血缘分析是通过血缘分析图对异常数据进行分析，查出问题的具体所在。影像性分析是提供由某个或某几个元模型变更影响的完整对象列表，它与血缘分析相反。

3.2 元数据管理系统的功能框架

元数据管理系统的功能框架包括数据源层、元数据获取层、元数据存储层、元数据服务接口层、元数据管理层和元数据应用层 6 个主要部分[5]，如图 2 所示。

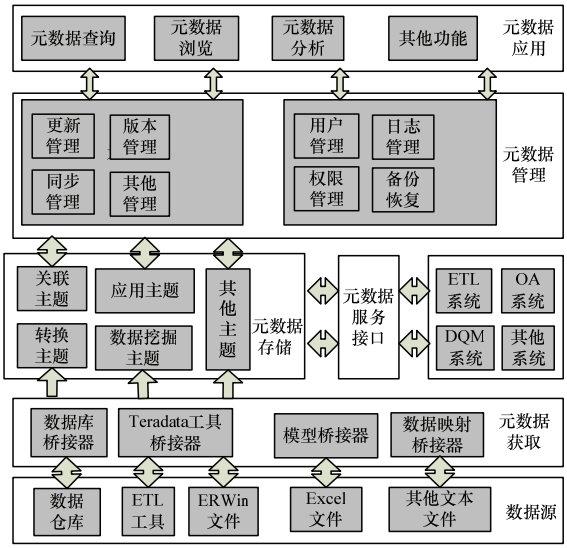

图 2 元数据管理系统的功能框架

元数据获取层将涉及的各子系统的元数据经过元数据桥接器导入到元模型中，元数据服务接口可以通过数据访问接口返回元数据中的数据内容，并生成其他数据系统需要的数据字典或提供其他应用的访问接口。元数据应用层提供元数据浏览、查询、分析的用户界面，提供与 ETL 系统、数据质量管理系统的数据交换机制。对各层的说明如下：

(1)元数据源层。元数据源层包括银行数据仓库涉及的数据仓库产品、数据挖掘工具、建立数据仓库过程中所需的数据信息(如 ERWin 文件、Excel 文件)等。

(2)元数据获取层。实现元数据源中各个系统的元数据抽取。元数据桥接器通过符合双方约定规范的接口或各个产品

提供的特定接口实现元数据的抽取，并把抽取出的元数据存入元数据存储部分中的元数据库。

(3)元数据存储层。实现元数据的存储，存储的元数据包括业务元数据和技术元数据，元数据按模型主题组织。存储库的逻辑模型设计须兼顾效率和模型的可扩展性与灵活性。

(4)元数据管理层。由元数据管理和系统管理 2 个部分构成。元数据管理实现元数据的更新管理、同步管理、版本管理等功能。系统管理实现用户管理、权限管理、日志管理、备份与恢复等功能。一些元数据管理部分的功能需要人工或半人工操作。

(5)元数据服务接口层。包括元数据对外的访问接口，包括 ETL、DQM、OA 系统或其他系统的服务接口，这些系统通过元数据服务接口部分访问元数据存储部分的元数据。该部分为其他用户或系统使用元数据提供了扩展方式。

(6)元数据应用层。提供元数据管理、技术、业务用户的访问。该部分实现元数据查询、元数据浏览、元数据分析等基本功能模块。

3.3 元数据管理系统的物理结构

本文中的元数据管理系统采用 B/S 架构，它运行在数据库服务器平台、应用服务器平台和用户终端平台上。普通技术用户和业务用户通过浏览器访问 Web 服务器。元数据管理系统的物理结构如图 3 所示。

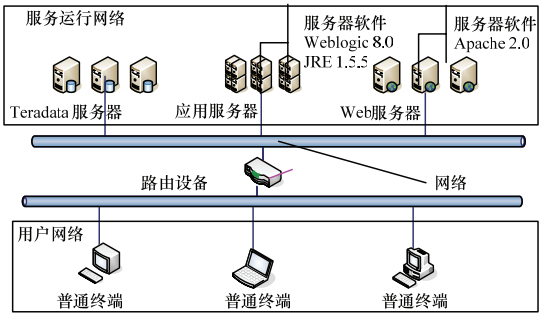

图 3 元数据管理系统的物理结构

对元数据管理系统各个平台说明如下：

(1)数据库服务器平台。实现元数据存储功能，数据库服务器上保存 MDS 引擎的数据库和其他相关数据库。

(2)应用服务器平台。实现元数据应用和元数据服务接口功能，在 Web 服务器上安装 Web 应用服务器和其他相关软件。

(3)用户终端平台。实现元数据获取和元数据管理功能，是元数据使用人员的工作终端。

4 结束语

元数据贯穿银行数据仓库系统中的各个环节，如数据抽取、转换和存储等。实现系统的各个处理单元由元数据驱动。因此，必须实施元数据的集中管理，提供一个集中的元数据全局视图，从而有效控制银行数据仓库系统重要信息数据的组成和转换，更好地管理数据仓库。近年来，数据质量控制的要求越来越高，使元数据管理的重要性更突出。本文提出的元数据管理系统在设计和实现过程中遵循 J2EE 的设计模式，具有良好的可扩展性和可维护性。

参考文献

[1] Arun S. Metadata Management: Past, Present, Future[J]. Science Direct Decision Support Systems, 2004, 37(1): 151-173.

[2] Missier P, Alper P, CorchoÓ. Requirements and Services for Metadata Management[J]. IEEE Internet Computing, 2007, 11(5): 17-25.

[3] 王 强, 刘东波, 王建新. 数据仓库元数据标准研究[J]. 计算机工程, 2002, 28(12): 123-125.

[4] OMG. Common Warehouse Metamodel Specification Version1.1 [S/OL]. (2003-03-02). http://www.omg.org/docs/formal/03-03-02.pdf.

[5] 杨鸿宾, 宋 明. 元数据管理平台总体架构设计研究[J]. 计算机系统应用, 2007, (11): 17-20.

编辑 陈 晖

(上接第 78 页)

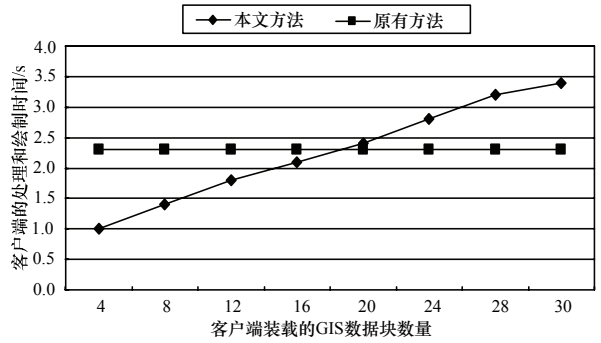

图 5 地图分块的平均处理时间

6 结束语

为了提高移动 GIS 服务的响应速度，实现基于查询服务的渐近计算，本文提出一种逻辑数据结构，讨论相关数据模式和处理方法。下一步工作主要包括数据的高效索引和缓冲管理，以支持快速分块操作并提高查询处理的效率，从而提高移动 GIS 服务组合的整体效率。

参考文献

[1] Saalfeld A. Topologically Consistent Line Simplification with the Douglas-peucker Algorithm[J]. Cartography and GIS, 1999, 26(1): 7-18.

[2] Wong E Y C. Efficient Management of XML Contents over Wireless Environment by Xstream[C]//Proceedings of the 19th ACM Symposium on Applied Computing. Nicosia, Cyprus: ACM Press, 2004: 1122-1127.

[3] 蔡海尼, 谢 军, 文俊浩, 等. 基于XML的地理数据集成研究及应用[J]. 计算机工程, 2008, 34(15): 77-79.

[4] Buttenfield B P. Transmitting Vector Geospatial Data Across the Internet[C]//Proceedings of Conf. on GIScience. Berlin, Germany: Springer-Verlag, 2002: 51-64.

[5] Zhou Min, Bertolotto M. A Data Structure for Efficient Transmission of Generalised Vector Maps[C]//Proc. of International Conference on Computational Science. Kraków, Poland: [s. n.], 2004: 948-955.

编辑 陈 晖