A Dissertation Submitted to Guangdong University of Technology for the Degree of Master of Engineering Science

Based on intelligent technology exchange platform for distributed database research and its implementation

Master Candidate: Lei Chong

Supervisor: rof.Lian Yingzhan

June 2010

Faculty of Automation
Guangdong University of Technology
Guangzhou, Guangdong, P.R. China, 510006

摘 要

随着 Internet 技术和通信技术的快速发展，政务办公系统和电子商务系统在政府和企业中广泛应用，为提高系统效率和功能，解决各个应用系统之间出现的“信息孤岛”问题，本文研究和设计了基于 XML 异构数据转换系统，为广东农村信息直通车工程门户应用系统的数据转换、数据共享以及数据的透明访问提供解决方案。

本文利用 XML 和 Java 技术，通过查阅数据转换的研究背景资料，研究并设计了异构数据转换模型，实现了异构数据库之间的数据转换。数据转换是从数据库到 XML 文档和从 XML 文档到数据库的双向转换。数据转换系统首先利用 XML 数据文档的 XML 模式生成数据库关系模式，然后根据生成的关系模式解析并提取 XML 数据文档中的数据，并将这些数据存储到相应的数据表中。

该系统利用 XML 的简单、自我描述性和平台无关性等优点，以 XML 文档作为中间过渡形式，实现数据由源数据库经由 XML 数据文档到目的数据库的转换。XML Schema 可以描述 XML 文档的结构，同时 XML Schema 还拥有丰富的基本数据类型和派生数据类型、自定义复杂数据类型的特点。本文使用 XML Schema 描述数据库的关系模式和 XML 数据文档的结构，规范和约束 XML 数据文档的有效性和合法性，为数据库关系模式和 XML 数据文档的结构之间建立映射。

该系统应用于广东农村信息直通车工程门户网站（Http://www.gdcct.gov.cn）数据资源用户提交数据与省平台的对接中，数据匹配准确，实现用户的要求，取得非常好的应用效果，系统在解决异构数据交换、数据共享等方面有一定的应用价值

关键词:XML ,Java ,数据交换，异构数据

ABSTRACT

With the rapid development of Intemet technology and communication technology ,the government office systems and ecommerce system were widely used by government and enterprise.To improve system efficiency and function and "information isolated island" problem, This paper study and design of heterogeneous data based on XML conversion system for Guangdong Through Train rural information system project portal application data conversion and Transparent data sharing and data access solutions.

By using XML and Java technology, through looking up data transformation and the background information, this paper research and design of heterogeneous data transformation model to achieve the conversion of data between heterogeneous databases, achieved the data transformation between heterogeneous databases, Data conversion from XML documents and databases to XML documents to the database from each other conversion.

The system using XML schema of the XML data document generation database relational model,and then according to the Relation between the mode of analysis and data extract XML data document,and stored the data in the corresponding data table.

The system uses XML, a simple, self-descriptive and platform independence, etc, In the form of XML documents used as transition, achieved the data from the source database via XML data files to the destination database conversion. XML Schema can describe the structure of XML documents, While XML Schema also has a wealth of basic data types and derived data types, custom features of complex data types. This article uses the XML Schema description of the relationship between the database model and the structure of XML data document, norms and constraints XML data document the effectiveness and legitimacy, relational schema for the database

and XML data mapping between the structure of the document.

The system is applied to train project in Guangdong Rural Information Portal(Http://www.gdcct.gov.cn)of its data resources and users to submit data in the province of the docking platform, accurate data matching, To achieve the users' requirements,and made a very good application effect, there is some value which the system in resolving heterogeneous data exchange, data sharing.

Keywords: XML, Java, Data exchange, Hetergeneous data

目 录

CONTENTS

第一章 绪论

1.1 课题的背景

中国电子信息产业发展研究院近日发布了《2005 中国信息化状况调查报告》。据悉，本次调查历时 3 个月，共涉及全国 1 万多家企事业单位和政府机构。 报告显示，2005 年的中国信息化状况发展良好，在行业和政府领域中，信息化都获得了长足的进步。政府是信息化应用普及的主力军，被调查的政府机构中 97.6%开展了电子政务应用。有 30%的受调查单位开始进行信息化的长期规划。在目前信息化应用技术中，“实时企业”被受访者认为是 2005 年中国信息化最有应用前景的技术，认同率高达 59%，其余依次为“网络和大容量存储”以及“协同商务”、“移动计算”等。在这些应用中，涉及的数据信息共享都将以数据交换平台为依托，只有实现了数据交换共享才能实现电子政务综合应用和提供高质量的服务。“数据是信息系统的根本”，对于电子政务、行业电子商务而言，不同的业务系统、不同的数据类型、不同的文件类型、不同的操作系统、不同的时间以及不同的阶段，引起信息的存储是分散孤立，信息共享成为系统未来必然趋势。解决该问题方式是研发一套安全、可靠、实时、高效的系统，提高数据利用率，降低数据存储成本，发挥数据最大效益。

2003 年，“中央一号”文件明确提出把农村信息化建设作为社会主义新农村建设的重要手段。广东省科技厅基于信息化是建设“三农”问题的必由之路的认识，提出了建设广东农村信息直通车工程，本文正是为广东农村信息直通车工程门户应用系统的数据转换、数据共享以及数据的透明访问而提出的。

1.2 课题的研究现状

一、数据交换系统研究现状

国内外有不少科研机构、IT 公司都对数据交换系统投入研究，如：IBM 公司有 IIS（Information Integrate System）、IBM MQ Series Integrator、IBM Lotus/Domino、微软 Exchange 产品等。归纳起来数据交换系统应用现状主要有以下的情况：

1、以文件交换为基础的数据交换，包括 Lotus Domino/Notes 等

2、狭隘电子数据交换（EDI），数据交换双方需协调统一数据交换格式，这是交换基础。

3、单一的数据库到数据库的交换。

二、国内外代表性产品

IBM MQSeries Integrator 是基于 MQSeries 开发的产品。它通过帮助业务应用跨平台地进行信息交换从而实现应用集成。MQSeries 集成器基本上实现了动态地处理和路由信息。它有 GUI 工具来设置一些业务规则，但使用起来不是很便捷，数据发送通过 MQSeries 消息来传送，其效率高，安全性好。MQSeries 集成器有一个开放式框架，信息格式要么定义在所提供的信息字典中，要么被定义成自我定义的 XML 信息。

IBM Lotus/Domino 产品是通过文件交换实现数据交换。它提供协作和人际交互必需的所有功能，并将包括消息收发、日程安排、在线感知、会议、聊天、工作流程、文档管理和内容管理等。通过 Lotus Workplace，这些功能可以任意组合，为拥有不同能力的各种用户提供服务，根据人员各自的职责需要实现协作和人际交互。客户可以选择他们希望激活的功能，并只为激活的功能支付费用。

微软 Exchange 2003 是 Microsoft 消息服务和协作服务器，旨在帮助企业更有效地进行通信。Exchange 2003 与 Microsoft Office Outlook® 2003 提供的丰富的客户端功能协同工作，可提供具有一流安全性和隐私性的移动、远程和桌面电子邮件访问；通过 Microsoft Windows Server™ 2003 提供服务降低了成本。

国内慧点科技设计了 DCI.DataExchanger。它是基于纯粹 WebService 技术建立的数据交换产品。系统采用数据中心和数据交换代理节点的结构，简化电子政务应用主体内部功能体之间、主体与主体之间所存在的复杂的相互关系，在代理节点上提供相应的服务来方便老应用系统的接入，并提

供一致的访问行为和接口。

三、发展趋势

数据交换的发展趋势如下：

1、数据交换的规模扩大

2、数据结构的表示复杂

3、数据交换的操作系统、网络环境等交换的环境复杂

4、数据交换涉及不同的地域数据迁移共享

1.3 课题的主要任务和意义

本课题的主要任务是在电子政务、电子商务领域信息整合需求基础上提出，解决异构信息系统的数据存储、数据共建、数据共享等问题，提出数据生命特征属性，开发适用于不同业务系统、不同数据类型、不同文件类型、不同操作系统、不同时间、不同阶段的基于 xml 技术的分布式数据交换平台，实现分布式应用业务系统数据的共建与共享。

其意义在于：

一、产品的应用领域：该平台是一个独立于应用行业的通用智能数据交换平台，可适用于各种行业的信息交换共享。

二、应用的市场需求：数据交换平台是当今多应用系统整合、异构数据库间信息交换系统。电子政务和企业应用整合（EAI）等方面需求尤为突出。

1、政府信息化领域。根据中国电子信息产业发展研究院发布的《2003 中国信息化状况调查报告》显示，在行业领域中，政府是信息化应用的主力军。信息化最前景的软件技术主要为“协同政务”、“数据交换”等。

2、企业应用整合领域。据统计 2005 年大型企业信息化累计投入平均为 6782.63 万元/户，约有 3.7%的大型企业信息化进入成熟阶段。未来两年建设以应用整合和数据交换为主。

三、市场规模与成长性分析

1、中国市场规模及成长性分析

（1）从市场需求的发展情况看，无论从电子商务、电子政务，都呈现

大规模推进与成长。经初步分析，信息化规划及咨询增长 5~6%,电子商务增长 20~25%，电子政务增长 33~40%。

（2）从目标市场的投入分析及预测看，2005 年上述项目标市场的投入会加大，加强电子商务应用呼声越来越高。项目投入增长的幅度会在30~38%左右。

（3）从产品及解决方案的综合竞争力看，经过一年半的研发、推广、讨论交流、会展、演示与评价，该平台在国内同类产品的竞争优势越来越明显。该平台在随着电子政务越来越冷静的过程中，越来越贴近电子政务实际需求，对于充分利用政府现有资源，打破“信息孤岛”， 该分布式智能数据交换平台具有极高适应性和弥合性。

2、预计市场占有分额：从目前来看，该平台独树一帜，具有鲜明的特点和优势，尤其是理念及思路等方面，突出在业界的形象与产品定位。截止目前，电子商务市场类似数据交换平台有三种类型及特点：

（1）基于 WEB 的信息交换平台：诸如 IBM 的 WEB 交换服务器、西安协同的 sincrofolw 等，其信息交换不能实现底层数据交换，具明显局限性；该平台市场占有份额约 21%；

（2）基于服务器的硬交换平台：诸如黎明网络的 I-SWITCH 等等，该交换系统需要借助专用的硬件交换设备，成本昂贵，效率低下，该类交换平台市场占有份额约在 10%左右；

（3）跨平台、跨数据、跨操作系统的分部式智能交换平台：该平台目前应用于国内食品行业等 11 个行业数据交换平台的市场占有额为 19%；在国内电子政务平台市场占有额预计在 2008 年底会达到 16%以上。

1.4 课题的主要研究工作

本课题以农业网站为例，在分析我国农业网站栏目的结构和特点的基础上，选择了网站资讯信息、供求信息、价格信息、农业科技四个栏目目标制定基于 XML 格式的交换标准,利用 XML 和 Java 技术，研究并设计了异构数据转换模型，以 XML 文档作为中间过渡形式,实现数据由源数据库经由 XML 数据文档到目的数据库的转换。

系统支持数据库直接查询，异构数据库之间表数据的交换乃至整个数据库的迁移。

1.5 论文的结构

本文是为广东农村信息直通车工程门户应用系统的数据转换、数据共享以及数据的透明访问提供解决方案，因此针对这一课题，本文的研究内容和结构如下：

第一章 绪论

提出课题的研究背景，研究现状，以及本课题的主要任务和意义，通过对要实现的系统的分析得出本课题的主要研究工作。

第二章 数据库接口技术

目前数据库接口技术众多，本章只选择其中具有代表性的几种接口技术：专用接口、ODBC、JDBC、ADO.NET。本文研究数据与 XML 之间的数据转换技术，其中要使用到数据库接口技术。

第三章 XML 介绍

介绍了 XML 的产生背景，以及它的概念以及特点，根据其特点得其应用范围，以及它的发展及其研究动态。

第四章 基于 XML 文档的转换方法

该章介绍了基于 XML 文档的两种转换方法:基于模板的数据转换方法和基于模型的数据转换方法。通过对系统实现要求的分析选择后一种实现方法。

第五章 数据转换系统的分析

通过对系统要实现的功能的分析得出数据转换系统模型，介绍了用于数据交换的 XML 文档，以及关系模式与 XML 模式的相互转换脚本。

第六章 系统的设计与实现

以全国农业网站数据交换系统中的实例说明它们的设计与实现，制定了基于 XML 格式的数据转换格式规范，详细描述了提取标准表单 XML 文档的方法，以及 XML 文档和关系数据库之间数据转换的转换脚本的编写方法，介绍了关键算法的实

第二章数据库接口技术

2.1 专用数据库接口

各个数据库开发商都为各自的产品提供了专用的数据库接口，如 SQL SQL Server 的 DB-Library 和 Oracle 的 OCI 等。专用数据库接口的特点是：访问数据库的速度快，但只适用于特定的数据库。通常实现的过程是：(1)登陆到数据库服务器；(2)接受并处理返回结果：(4)处理接口发出的错误和服务器的消息；(5)关闭与服务器的连接并释放缓冲。

2.2 ODBC

ODBC(Open Database Connectivity，开发数据库互联)是微软公司开放服务结构(WOSA，Windows Open Services Architecture)中有关数据库的一个组成部分，它建立了一组规范，并提供了一组对数据库访问的标准 API（应用程序编程接口）。这些 API 利用 SQL 来完成其大部分任务。ODBC 本身也提供了对 SQL 语言的支持，用户可以直接将 SQL 语句送给 ODBC[1]。

一个基于 ODBC 的应用程序对数据库的操作不依赖任何 DBMS，不直接与 DBMS 打交道，所有的数据库操作由对应的 DBMS 的 ODBC 驱动程序完成。也就是说，不论是 FoxPro、Access，MYSQL 还是 Oracle 数据库，均可用 ODBC API 进行访问。由此可见，ODBC 的最大优点是能以统一的方式处理所有的数据库。

一个完整的 ODBC 由下列几个部件组成：

1.应用程序(Application)。

2.ODBC 管理器(Administrator)。该程序位于 Windows 95 控制面板(Control Panel)的 32 位 ODBC 内，其主要任务是管理安装的 ODBC 驱动程序和管理数据源。

3.驱动程序管理器(Driver Manager)。驱动程序管理器包含在ODBC32.DLL 中，对用户是透明的。其任务是管理 ODBC 驱动程序，是ODBC 中最重要的部件。

4.ODBC API。

5.ODBC 驱动程序。是一些 DLL,提供了 ODBC 和数据库之间的接口。

6.数据源。数据源包含了数据库位置和数据库类型等信息，实际上是一种数据连接的抽象。

应用程序要访问一个数据库，首先必须用 ODBC 管理器注册一个数据源，管理器根据数据源提供的数据库位置、数据库类型及 ODBC 驱动程序等信息，建立起 ODBC 与具体数据库的联系。这样，只要应用程序将数据源名提供给 ODBC，ODBC 就能建立起与相应数据库的连接[1]。

在 ODBC 中，ODBC API 不能直接访问数据库，必须通过驱动程序管理器与数据库交换信息。驱动程序管理器负责将应用程序对 ODBC API 的调用传递给正确的驱动程序，而驱动程序在执行完相应的操作后，将结果通过驱动程序管理器返回给应用程序。

在访问 ODBC 数据源时需要 ODBC 驱动程序的支持。用 Visual C++ 5.0 安装程序可以安装 SQL Server、 Access、 Paradox、 dBase、FoxPro、 Excel、 Oracle 和 Microsoft Text 等驱动程序. 在缺省情况下，VC5.0 只会安装 SQL Server、 Access、 FoxPro 和 dBase 的驱动程序. 如果用户需要安装别的驱动程序，则需要重新运行 VC 5.0 的安装程序并选择所需的驱动程序。

ODBC 使用层次的方法来管理数据库，在数据库通信结构的每一层，对可能出现依赖数据库产品自身特性的地方，ODBC 都引入一个公共接口以解决潜在的不一致性，从而很好地解决了基于数据库系统应用程序的相对独立性,这也是 ODBC 一经推出就获得巨大成功的重要原因之一。

2.3 JDBC

JDBC（Java Data Base Connectivity,java 数据库连接）是一种用于执行 SQL 语句的 Java API,可以为多种关系数据库提供统一访问，它由一组用 Java 语言编写的类和接口组成。JDBC 为工具/数据库开发人员提供了一个标准的 API，据此可以构建更高级的工具和接口，使数据库开发人员能够用纯 Java API 编写数据库应用程序，同时，JDBC 也是个商标名[1][2]。

有了 JDBC，向各种关系数据发送 SQL 语句就是一件很容易的事。换言之,有了 JDBC API,就不必为访问 Sybase 数据库专门写一个程序，为访问 Oracle 数据库又专门写一个程序，或为访问 Informix 数据库又编写另一个程序等等，程序员只需用 JDBC API 写一个程序就够了，它可向相应数据库发送 SQL 调用。同时，将 Java 语言和 JDBC 结合起来使程序员不必为不同的平台编写不同的应用程序，只须写一遍程序就可以让它在任何平台上运行，这也是 Java 语言“编写一次，处处运行”的优势。

Java数据库连接体系结构是用于Java应用程序连接数据库的标准方法。JDBC 对 Java 程序员而言是 API，对实现与数据库连接的服务提供商而言是接口模型。作为 API，JDBC 为程序开发提供标准的接口，并为数据库厂商及第三方中间件厂商实现与数据库的连接提供了标准方法。JDBC 使用已有的 SQL 标准并支持与其它数据库连接标准，如 ODBC 之间的桥接。JDBC 实现了所有这些面向标准的目标并且具有简单、严格类型定义且高性能实现的接口。

Java 具有坚固、安全、易于使用、易于理解和可从网络上自动下载等特性，是编写数据库应用程序的杰出语言。所需要的只是 Java 应用程序与各种不同数据库之间进行对话的方法。而 JDBC 正是作为此种用途的机制。

JDBC 扩展了 Java 的功能。例如，用 Java 和 JDBC API 可以发布含有 applet 的网页，而该 applet 使用的信息可能来自远程数据

库。企业也可以用 JDBC 通过 Intranet 将所有职员连到一个或多个内部数据库中（即使这些职员所用的计算机有 Windows、 Macintosh 和 UNIX 等各种不同的操作系统）。随着越来越多的程序员开始使用 Java 编程语言，对从 Java 中便捷地访问数据库的要求也在日益增加。

MIS 管理员们都喜欢 Java 和 JDBC 的结合，因为它使信息传播变得容易和经济。企业可继续使用它们安装好的数据库，并能便捷地存取信息，即使这些信息是储存在不同数据库管理系统上。新程序的开发期很短。安装和版本控制将大为简化。程序员可只编写一遍应用程序或只更新一次，然后将它放到服务器上，随后任何人就都可得到最新版本的应用程序。对于商务上的销售信息服务， Java 和 JDBC 可为外部客户提供获取信息更新的更好方法。

2.4 ADO.NET

ADO.NET 的名称起源于 ADO(ActiveX Data Objects),这是一个广泛的类组,用于在以往的 Microsoft 技术中访问数据.之所以使用 ADO.NET 名称,是因为 Microsoft,希望表明,这是在.NET 编程环境中优先使用的数据访问接口.

它提供了平台互用性和可伸缩的数据访问。ADO.NET 增强了对非连接编程模式的支持,并支持 RICH XML.由于传送的数据都是 XML 格式的，因此任何能够读取 XML 格式的应用程序都可以进行数据处理。事实上，接受数据的组件不一定要是 ADO .NET 组件，它可以是基于一个 Microsoft Visual Studio 的解决方案,也可以是任何运行在其它平台上的任何应用程序[1][2]。

ADO.NET 是一组用于和数据源进行交互的面向对象类库。通常情况下,数据源是数据库,但它同样也能够是文本文件、Excel 表格或者 XML 文件。ADO.NET 允许和不同类型的数据源以及数据库进行交互。然而并没有与此相关的一系列类来完成这样的工作。因为不同的数据源采用不同的协议，所以对于不同的数据源必须采用相应的协议。一些老式

的数据源使用 ODBC 协议，许多新的数据源使用 OleDb 协议，并且现在还不断出现更多的数据源，这些数据源都可以通过.NET 的 ADO.NET 类库来进行连接。

ADO.NET 提供与数据源进行交互的相关的公共方法，但是对于不同的数据源采用一组不同的类库。这些类库称为 Data Providers，并且通常是以与之交互的协议和数据源的类型来命名的。

总之 ADO.NET 是与数据源交互的.NET 技术。有许多的 Data Providers，它将允许与不同的数据源交流——取决于它们所使用的协议或者数据库。然而无论使用什么样的 Data Provider，你将使用相似的对象与数据源进行交互。SqlConnection 对象管理与数据源的连接。SqlCommand 对象允许你与数据源交流并发送命令给它。为了对进行快速的只“向前”地读取数据，使用 SqlDataReader。如果想使用断开数据，使用 DataSet 并实现能进行读取或者写入数据源的 SqlDataAdapter。

2.5 本章小结

本章主要介绍了诸多数据库接口中的具有代表性的几种：专用接口、ODBC、JDBC、ADO.NET，分析了专用数据库的特点访问速度快，但是由于其只适用于特定数据库，因此移植性差，无法适用系统的设计要求，然后详细介绍了几种通用的数据库接口，为后面的数据库接口的选择提供了理论依据

第三章 XML介绍

3.1 XML的产生及特点

XML同HTML一样，都来自Standard Generalized Markup Language，即标准通用标记语言，简称SGML。早在Web未发明之前，SGML就早已存在。正如它的名称所言，SGML是一种用标记来描述文档资料的通用语言，它包含了一系列的文档类型定义（简称DTD），DTD中定义了标记的含义，因而SGML的语法是可以扩展的。SGML十分庞大，既不容易学，又不容易使用，在计算机上实现也十分困难。鉴于这些因素，Web的发明者--欧洲核子物理研究中心的研究人员根据当时(1989年)计算机技术的能力，提出了HTML语言[2][3]。

HTML只使用SGML中很小一部分标记，例如HTML 3.2定义了70种标记。为了便于在计算机上实现，HTML规定的标记是固定的，即HTML语法是不可扩展的，它不需包含DTD。HTML这种固定的语法使它易学易用，在计算机上开发HTML的浏览器也十分容易。正是由于HTML的简单性，使Web技术从计算机界走向全社会，走向千家万户，Web的发展如日中天。

近年来，随着Web的应用越来越广泛和深入，人们渐渐觉得HTML不够用了，HTML过于简单的语法严重地阻碍了用它来表现复杂的形式。尽管HTML推出了一个又一个新版本，已经有了脚本、表格、帧等表达功能，但始终满足不了不断增长的需求。另一方面，这几年来计算机技术的发展也十分迅速，已经可以实现比当初发明创造HTML时复杂得多的Web浏览器，所以开发一种新的Web页面语言既是必要的，也是可能的。

有人建议直接使用SGML作为Web语言，这固然能解决HTML遇到的困难。但是SGML太庞大了，用户学习和使用不方便尚且不说，要全面实现SGML的浏览器就非常困难，于是自然会想到仅使用SGML的子集，使新的语言既方便使用又实现容易。正是在这种形势下，Web标准化组织W3C建议使用一种精简的SGML版本--XML应运而生了。

XML 有以下几种特点[4][5]：

(1) 开放的国际化标准

XML 是 W3C 正式批准的，它完全可用于 Web 和工具的开发。XML 具有标准的名域说明方法，支持文档对象模型标准、可扩展链接语言标准和 XML 指针语言标准。使用 XML 可以在不同的计算机系统间交换信息，而且还可以跨越国界和超越不同文化疆界交换信息。XML 种还包括可扩展格式语言 XSL(Extensible Style Language)和可扩展链接语言 XLL(Extensible Linking Language)使得 XML 显示和解析更加方便快捷。

(2) 高效可扩充

XML 允许程序自行定义标记来满足需要。同样一个行业或某一特定人群，也可以指定在自己范围内的通用标记集，这样 XML 可以轻松地适应每一个领域而无需对语言本身做大修改。另外，XML 的数据定义与数据本身分离而独立存在，这样使 XML 的标记集不致日益扩大。XML 支持复用文档片段，使用者可以发明和使用自己的标签，也可以与他人共享，可延伸性大。在 XML 中，可定义一组无限量的标准，可以有效地进行 XML 文件的扩充。

(3) 良好的移植性能

XML 语言可以定义各种数据，比如说文本、图像、声音等。这些数据往往有很多种不同的格式使得数据不能在各系统之间交流，或者可以使用额外的转换软件来实现跨平台的交流。XML 的这个特点使得只要交换数据的系统都能处理一种格式的文件即 XML 文档，就能处理由 XML 标注的各种数据，从而实现了不同格式数据的跨平台交换。

(4) 良好的自描述性

XML 有许多部分，但是只需要了解其中的三个就可以了解它是怎样工作的。它们是：文档类型定义(Document Type Definition,DTD),也就是 XML 的布局语言：可扩展的样式语言(Extensible Style Language:XSL),也就是 XML 的样式表语言：以及可扩展链接语言(Extensible LinkLanguage:XLL)。因而 XML 文档是自描述的。XML 文档良好的自描述性不仅使文档能被人读懂，还可以被不同的应用程序识别、分析和处理。

总之 XML 不仅能满足不断增长的网络应用需求，同时能够确保在通过

网络进行交互合作时，具有良好的可靠性与互操作性。它是一种自我描述的定义语言，用户自己可以定义标记来描述文件中的任何数据元素，从而突破了HTML固定标记集合的约束，使文件的内容更丰富更复杂并组成一个完整的信息体现。由于这些优点，XML 已经进化成了一个电子商务和信息交换的全球平台。

3.2 XML 的基本概念

3.2.1 XML 的基本语法结构

一个 XML 文件通常包含文件头和文件体两大部分

1. 文件头

XML 文件头由 XML 声明与 DTD 文件类型声明组成。其中 DTD 文件类型声明是可以缺少的，关于 DTD 声明将在后续的内容中介绍，而 XML 声明是必须要有的，以使文件符合 XML 的标准规格。

在前面的 Flowers.xml 文件中的第一行代码即为 XML 声明：

<?xml version="1.0" encoding="gb2312"?>

其中：

“<?”代表一条指令的开始，“?>”代表一条指令的结束；

“xml”代表此文件是 XML 文件；

“ version="1.0" ”代表此文件用的是 XML1.0 标准；

“ encoding="gb2312" ” 代表此文件所用的字符集，默认值为 Unicode，如果该文件中要用到中文，就必须将此值设定为 gb2312。XML 声明必须出现在文档的第一行。

2. 文件体

文件体中包含的是 XML 文件的内容，XML 元素是 XML 文件内容的基本单元。从语法讲，一个元素包含一个起始标记、一个结束标记以及标记之间的数据内容。

XML 元素与 HTML 元素的格式基本相同，其格式如下：

<标记名称 属性名 1="属性值 1" 属性名 1="属性值 1" ……>内容</

标记名称>

所有的数据内容都必须在某个标记的开始和结束标记内，而每个标记又必须包含在另一个标记的开始与结束标记内，形成嵌套式的分布，只有最外层的标记不必被其他的标记所包含。最外层的是根元素(Root)，又称文件(Document)元素，所有的元素都包含在根元素内。

在前面的 Flowers.xml 文件中，根元素就是<Flowers>，根元素必须而且只能有一个，在该文件有三个<Flower>子元素，这样的元素可以有多个。

3.2.2 文件合法性检验

一个结构良好的 XML 文档仍可能是非法的，如：

```
<商品>
    <名称>法拉利</名称>
    <单价>10</单价>
    <单价>20</单价>
<商品/>
```

上面文档是结构良好的，但出现了两次的<单价>，显然提供的数据不明确、不合逻辑。为了避免上述情况，W3C 规定 XML 文档除了满足 well-formed 外，还必须是 valid(合法的)，并定义了一系列规则来规范之，即 DTD 或 XML Schema。

1.DTD

文档类型定义(DTD)是一套关于标记符的语法规则。它指出在文档中使用哪些标记符，它们应该按什么次序出现，哪些标记符可以出现于其它标记符中，哪些 标记符有属性等等。DTD 原来是为使用 SGML 开发的，它可以是 XML 文档的一部分，但是它通常是一份单独的文档或者一系列文档[6]。XML 本身并没有一个通用的 DTD，想使用 XML 进行数据交换的行业或组织可以定义它们自己的 DTD。

DTD 文档与 XML 文档有比较大的不同，而且 DTD 也并不能完全满足 XML 自动化处理的要求，例如不能很好实现应用程序不同模块间的相互协调，缺乏对文档结构、属性、数据类型等约束的足够描述等等。

2.XML Schema

XML Schema 已经成为 W3C 的正式推荐标准，并有替代 DTD 的趋势。

XML Schema是用一套预先规定的XML元素和属性创建的，这些元素和属性定义了文档的结构和内容模式。相应的一套精巧的规则指定了每个Schema元素或者属性的合法用途。如果违反这些规则解析器就会拒绝解析你的Schema以及任何同它相联系的文档[7][8]。

XML Schema事实上也是XML的一种应用，也就是说XML Schema的格式与XML的格式是完全相同的，因此XML具有的优点它都具有。XML Schema比DTD支持更多的数据类型。因此，XML Schema有更大的发展前景。

3.XML的命名空间

当我们在一个XML文档中使用他人的或者多个DTD文件，就会出现这样的矛盾：因为XML中标识都是自己创建的，在不同的DTD文件中，标识名可能相同但表示的含义不同，这久可能引起数据混乱[9][10]。比如在一个文档<table>woodtable</table>中<table>表示桌子，而在另一个文档<table>namelist</table>中<table>表示表格。如果我们需要同事处理这两个文档，就会发生名字冲突。为了解决这个问题，我们引进了名称空间这个概念。

名称空间通过给标识名称加一个网址(URL)定位的方法来区别这些名称相同的标识。名称空间在XML文档的开头部分声明，声明的语法如下：

<document xmlns:yourname='URL'>

其中yourname是有你定义的namespaces的名称，URL就是名字空间的网址。假设上面的“桌子<table>”文档来自http://www.163.com,我们就可以声明为：

<document xmlns:zhuozi='http://www.163.com'>

然后在后面的标识中使用定义好的名字空间：

<zhuozi:table>wood table</table>

这样就将这两个<table>区分开来。注意的是：设置URL并不是说这个标识真的要到那个网址去读取，仅仅作为一种区别的标志而已。

3.2.2 文档对象驱动模型

迄今为止，我们一直将 XML 作为这样一种工具，用它描述数据的结果是可供人阅读的文档。其实，XML 最令人称赞的功能恐怕要算是它表现信息结构的能力，即文档各个部分之间的关系以及它们如何组织成为一个具有确定意义的整体--正如数据库中的表能够描述各部分数据的关系。如结构良好规则和更为严格的 DTD 定义所指出的，XML 文档内各个元素之间不是简单的前后次序关系，而是具有严格的嵌套、依赖关系。XML 文档作为一个具有确定意义的信息整体，其部分语义正是通过这种结构关系得以体现[11][12]。

在 DOM 下，程序所面对的 XML 文档不是一个文本流，而是一棵对象树。程序可以方便地提取或修改任意对象或它的属性，这些属性可以是与当前对象对应的元素的子元素列表，也可以是它所包含的文本。所有 XML 文档具有以下特征：它的所有元素分层嵌套形成一个树形结构。因此，我们不仅可以简单地把一个 XML 文件看成是一个文本文件，而且还可以看成如下的标记树。在这棵标记树中，每一个 XML 元素对应一个树节点，所有子节点都依次嵌套于它的父节点[13][14]。

文档对象模型就是这样一个结构化文档编程接口（ API ），它定义了文档的逻辑结构以及访问和操纵文档的方法。使用 DOM 模型，程序员可以方便地创建文档、导航其结构，或增加、修改、删除、移动文档的任何成份。DOM 标准的出现大大简化了结构化文档在编程环境中的处理。例如，利用 DOM 可以很方便地写出一个图形化的 XML 文档编辑程序：用图形表示文档中各个元素以及它们之间的关系，用户可以非常直观地编辑文档结构、元素、属性。

在 DOM 中主要有以下三个对象：

(1)XML 文档对象 XML 文档既是一种对象，同时又代表整个 XML 文档。它由根元素和子元素组成。

(2)XML 节点对象 XML 节点对象代表的是 XML 文档内部的节点，如元素、注释、名字空间等。

(3)XML 节点列表 XML 文档模块列表代表了节点的集合。

利用 DOM，开发人员可以动态地创建 XML 文档，遍历结构，添加、修改、删除内容等。其面向对象的特性，使人们在处理 XML 解析相关的事物时节省大量的精力，是一种符合代码重用思想的强有力编程工具。

3.2.3 事件驱动模型

SAX 是 simple API for XML 的缩写，它不从属于任何官方机构，可以说 SAX 标准是在“民间”形成的，但是它却得到了广泛的认同，几乎所有的解析器都支持 SAX 标准。

因为 SAX 基于事件驱动，所以有必不可少的两部分：解析器和事件响应。前者负责解析，即读取 XML 文档，分析后调用相应的事件响应，对得到的 XML 数据进行处理。而且，SAX 还是以一种流文件的方式处理 XML 文档，也就是每读取一段就可以分析，无需将所有的数据全部读完，这样就使得 SAX 具有很高的解析效率。

不同的事件被事件管理器调用不同的响应器响应。SAX 只提供了相应的解析器、事件处理器接口(org.xml.sax)。然后由具体的解析器提供方负责实现这些接口，其中最重要的是 XMLReader 接口（负责解析），处理器程序则由我们自己编写。

当我们编写事件响应时，一般要实现 ContentHander 接口，然后使用 XMLReader 对象的 setContentHandler()方法注册一个 ContentHandler 实例。解析器就会通过这个事件实例来处理事件。

总之，SAX 是基于事件驱动，以流的方式加载、分析的。解析器在解析过程中触发一系列事件，激活相应的回调方法，对信息进行处理。很明显，SAX 不需要全将数据存放在内存，因此解析耗时少、内存需求低；但是我们不能直接对 XML 文档进行修改[15][16]。

3.3 XML 的应用分类

总的说来 XML 的应用可以分为四类：

（1）应用于客户需要与不同的数据源进行交互时。数据可能来自不同的数据库，他们都有各自不同的复杂格式。但客户与这些数据库间只通过一种标准语言进行交互，那就是 XML。由于 XML 的自定义性及可扩展性，它足以表达各种类型的数据。客户收到数据后可以进行处理，也可以在不同数据库间进行传递。总之，在这类应用中，XML 解决了数据的统一接口问题。但是，与其他的数据传递标准不同的是，XML 并没有定义数据文件中数据出现的具体规范，而是在数据中附加 tag 来表达数据的逻辑结构和含义。这使 XML 成为一种程序能自动理解的规范。

（2）应用于将大量运算负荷分布在客户端，即客户可根据自己的需求选择和制作不同的应用程序以处理数据，而服务器只须发出同一个 XML 文件。仍以上例为论，如按传统的"客户/服务器"工作方式，客户向服务器发出不同的请求，服务器分别予以响应，这不仅加重服务器本身的负荷，而且网络管理者还须事先调查各种不同的用户需求以做出相应不同的程序，但假如用户的需求繁杂而多变，则仍然将所有业务逻辑集中在服务器端是不合适的，因为服务器端的编程人员可能来不及满足众多的应用需求，也来不及跟上需求的变化，双方都很被动。应用 XML 则将处理数据的主动权交给了客户，服务器所作的只是尽可能完善、准确地将数据封装进 XML 文件中,正是各取所需、各司其职。XML 的自解释性使客户端在收到数据的同时也理解数据的逻辑结构与含义，从而使广泛、通用的分布式计算成为可能。

（3）应用于将同一数据以不同的面貌展现给不同的用户。这一应用也可在上例中体现出来。它又类似于同一个剧本，我们却可以用电视剧、电影、话剧、动画片等不同形式表现出来。这一应用将会为网络用户界面个性化、风格化的发展铺平道路。

（4）应用于网络代理对所取得的信息进行编辑、增减以适应个人用户的需要。有些客户取得数据并不是为了直接使用而是为了根据需要组织自己的数据库。比方说，教育部建立一个庞大的题库，考试时将题库中的题目取出若干组成试卷，再将试卷封装进 XML 文件，接下来便是最精彩部份，在各个学校让其通过一个过滤器，滤掉所有的答案，再发送到各个考生面前，未经过滤的内容则可直接送到老师手中，当然考试过后还可以再传送

一份答案汇编。此外，XML 文件中还可以包含进诸如难度系数、往年错误率等其他相关信息，这样只需几个小程序，同一个 XML 文件便可变成多个文件传送到不同的用户手中。

3.4 XML 的发展与研究动态

当前 Internet 已成为业界不可缺少的生命线，信息系统不管处于何种形态都要考虑接入 Internet，以 Microsoft、IBM、Oracle 等为首的大公司都在以 XML 数据格式为标准进行开发，XML 已经成为实现企业间数据交换的核心技术，活跃在电子商务等应用领域。1999 年 3 月 Microsoft 发表了 XML 数据交换的标准“Biztalk”；Oracle 公司发表的基于 XML 消息的连接中间件也进入市场，其 ERP 软件包 Oracle Application 和 Oracle Application Service 都使用了 XML 数据。IBM 公司的电子商务软件 Net。commerce 增加了对使用 XML 的企业进行连接的 Commerce Integration 软件。

事实上，目前 XML 应用的并不很多。阻碍 XML 在 Web 上的应用有两个原因。第一，XML 是非常新的标准，还处于不成熟的阶段。它只不过获得了 World Wide Web Consortium 表面上的推荐，还必须等待其他一些基本技术的支持；第二，因为参与开发工作的公司较多，所以除非完成标准化工作以及纵向市场的开发，否则很难实现实用的目的。人们正在讨论其标准和开发工具。像 Adobe Systems、IBM、Microsoft、Netscape 以及 Sun Microsystems 这些公司都正在忙于提出各种基于 XML 的标准，并以各自的样式风格在现有的产品中支持 XML。结果导致 XML 在 Web 和商业上相互冲突，使得其实实施变得越来越不那么通畅。

然而在 Internet 的发展前沿，随着相关标准的确立，XML 必然成为当今的热点，同时，基于它的各种语言也将相继出现。使 XML 技术和标准越来越成熟。

当前 XML 主要应用领域体现在四个方面[17][18]：

1.企业电子商务网。在企业间正在从原有的广泛实用的 EDI 技术转向 XML/EDI.

2.知识管理。XML 可以对各种文档和资料进行真正实用的知识管理。使用 XML 的文档结构化和文档含义化功能，可以统一进行多项目索引管理，而且新老文档可以混合实用。在该领域，XML 必成为季候的发展主流。

3.文档管理。在文档管理中，XML 的最大优势是可以直接在 WWW 显现 XML 数据，工具种类和数量丰富，实用 Unicode 代码不依赖于工具，在文档中指定 URL 能直接利用 Internet 数据。用 XML 管理文档数据，面向多种形态媒体可输出同一数据，易于维护文档，降低成本。

4.实现企业间自动化处理。目前在美国活跃着一个联合组织称为 RossttaNet，其目标是使用 XMl 技术进行供应链大改造，实现企业间电子目录的分配、更新、以及市场信息和在库信息的共享，彻底提高企业的效率。

3.5 本章小结

本章简单的介绍了 XML 的产生，特点，及其基本概念，根据 XML 自身的特点，对 XML 的应用作了分类，并对 XML 今后的发展与研究动态作了详细的介绍

第四章 基于 XML 文档的数据转换方法

目前企业内的数据仍主要以关系型数据、OCR 数据、文本数据等格式保存，因此企业间要实现 XML/EDI，必须要解决 XML 文档与其它数据格式的转换问题。本章主要研究基于 XML 文档的数据转换方法。

XML 文档属于半结构化的数据，将它与结构化数据（关系数据库和 EDI 数据）或非结构化数据进行转换的时候，关键问题是建立不同结构层次之间的映射关系。即如何将 XML 文档的结构与其他格式数据的结构对应起来。根据映射关系的建立方式不同，我们可以得到两种数据转换方法：基于模板的转换方法和基于模型的转换方法。

4.1 基于模板的数据转换方法

基于模板的转换方法并不事先定义好 XML 文档与其他数据之间的映射关系，而是在 XML 文档中嵌入一些可执行的指令。这些指令在转换过程中被系统所识别和执行，执行的结果被替换到指令所在的位置，从而生成目标 XML 文档。以数据库数据为例，为了从数据库中获取航班信息，并将航班信息用 XML 文档表示出来，我们可以定义如下的模板：

```
<?xml version="1.0"?>
<FlightInfo>
<Intro>
        The following flights have available seats:
</Intro>
<Select Stmt>
        SELECT Airline,FltNumber,Depart,Arrive FROM Flights
</SelectStmt>
<Conclude>
 We hope one of these meets your needs
```

```
</Conclude>
</FlightInfo>
```

要生成 XML 文档时，系统扫描整个模板，当遇到<SelectStmt>指令的时候，系统识别出这是一个可执行指令，于是调用指令执行后的结果：

```
<?xml version="1.0"?>
<FlightInfo>
<Intro>
The following flights have available seats:
</Intro>
<Flights>
 <ROW>
<Airline>ACME</Airline>
<FltNumber>123</FltNumber>
<Depart>Dec 12, 1998 13:43</Depart>
<Arrive>Dec 13, 1998 01:21</Arrive>
</Row>
...
</Flights>
<Conclude>
  we hope one of these meets your needs
</Conclude>
</FlightInfo>
```

图 4-1 给出了基于模板的转换方法的框架图

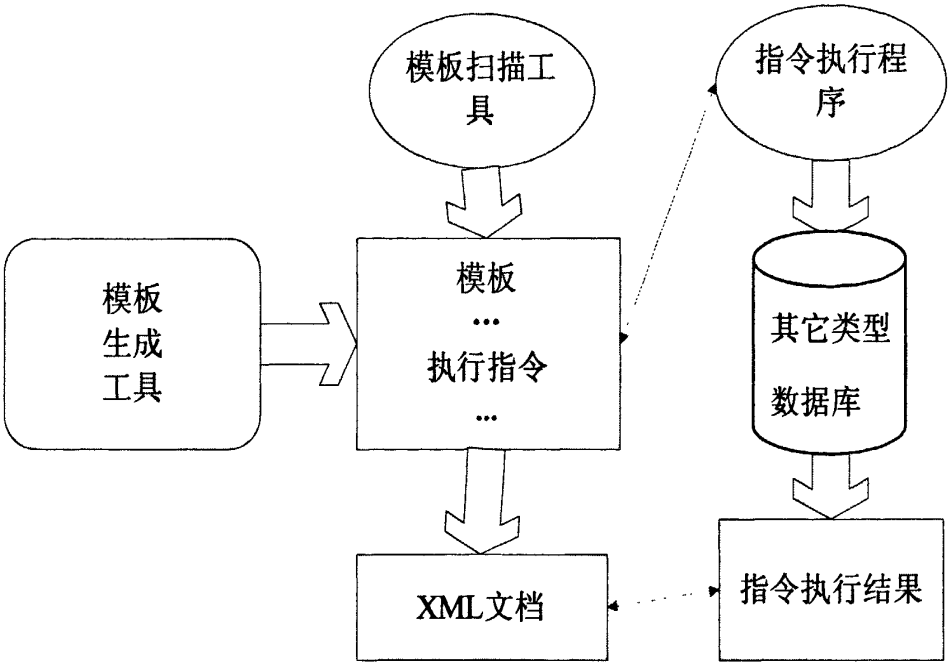


图 4-1 基于模板的转换框架图

Fig.4-1 Template-based framework for graph transformation

基于模板的转换方法的好吃在于转换的步骤比较简单，只要给出模板，就可以快速地生成相应的 XML 文档。不足之处在于，它只合适将其他类型的数据转换为 XML 文档，对于反向的转换就无能为力了。此外，基于模板的转换方法关键是要生成大量合理的模板，为此，系统需要为用户提供一套生成模板的工具，以及相应的指令执行程序。对于数据库数据，可以借用数据库管理系统方便地生成指令执行程序；而对于文本数据和 OCR 数据，指令执行程序的编写也是需要大量的工作的[20]。

4.2 基于模型的数据转换方法

基于模型的转换方法用事先定义好的数据模型来映射 XML 文档结构与其他格式数据的结构之间的关系。以数据库为例，一个最为简单的模型就是将文档结构定义为如下的模型：

```
<database>
  <table>
    <row>
```

```
    <column1>...</column1>
    <column2>...</column2>
    ...
  </row>
    ...
  </table>
  ...
</database>
```

把数据库数据转换成 XML 文档时，只要把一个表或一个查询结果的数据插入到相应位置即可；而把 XML 文档数据转换成数据库数据时，只要把内容插入到相应的表中即可。

另一种常用的模型是将 XML 文档的结构定义为一棵数据库对象树，根据规则将文档的层次结构转换为树状结构（通常是把文档中的元素定义为树的节点）。这种模型对于 XML 文档与面向对象数据库和层次数据库之间的转换是非常便利的。当与关系数据库进行转换时，可以利用传统的“对象——关系”映射技术来实现。

图 4-2 给出了基于模型的转换方法的框架图。

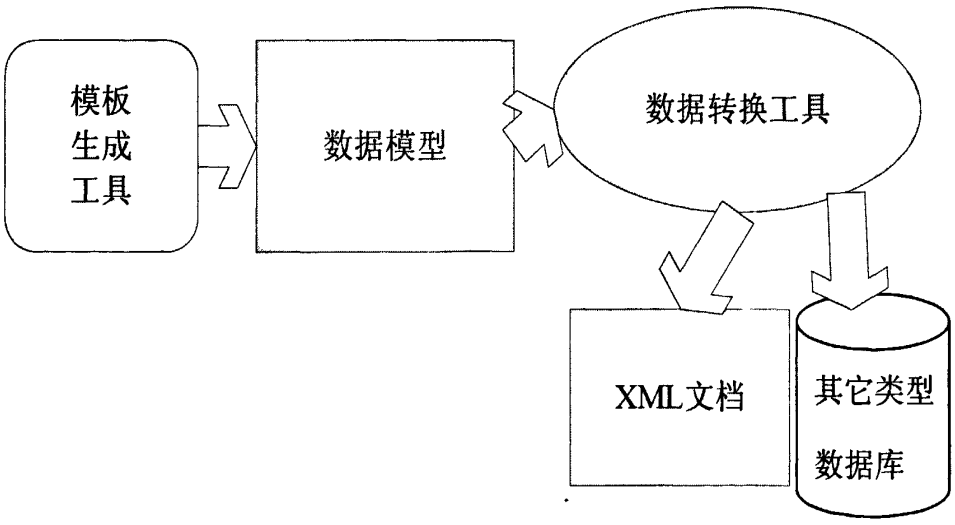


图 4-2 基于模板的转换框架图

Fig.4-2 Template-based framework for graph transformation

基于模型的转换方法由于有数据模型的支持，转换工作相对比较简单，

并且可以完成XML数据与其他格式数据之间的双向转换。但是模型的引入也使得XML文档的结构受到了一些限制，一个XML文档必须符合模型所规定的结构，才能将XML文档转换成其他类型的数据，而从其他类型数据转换得到的XML文档也具有某种结构特点。所以基于模型的转换方法的关键在于设计一个灵活的映射模型，使得对XML文档结构的限制尽量的少。

4.3 基于元素树的查询

基于元素树的转换方法实际是一种基于模型的转换方法，该方法首先创建元素树以及元素树节点与其他类型数据之间的映射关系，然后在元素树和映射关系的基础上完成数据转换[23]。

4.3.1 元素树

一棵元素树就是一个DTD所包含的元素之间的关系树，它的构成如下：

●元素树的每一个节点对应于DTD中的一个元素。

●每个节点包含以下主要信息：属性列表，子元素列表和其他信息。

●属性列表包含了该元素的所有属性，每个属性是一个三元组(属'眭名，属性值，属性类型)。

●子元素列表包含了该元素的所有子元素，每个子元素对应一个新的节点。

●其他信息包括：父亲节点、元素内容模式、元素内容出现次数和JL素文本内容等。一棵元素树在某些情况下足非完全的，即当有些元素的内容模式中包含子内容模式时，系统将无法确定子内容模式定义的内容应该如何出现。此时，需要插入内容模式节点，说明情况，然后根据实际的数据内容来确定出现哪个内容，应该出现多少次。所谓内容模式节点是一种特殊节点，它是为了体现子内容模式而建立的， 个内容模式节点的子节点是同属于该子内容模式的所有元素所对应的节点。例如下而的定义的元素fatfler，其内容模式是序列列表，即son1、son2或son3、son4是按顺序出现在XML文档中的。在father元素的内容模式中包含了一个子内容

模式，该子内容模式是内容粒子的选择列表，即 son2 和 son3 只能在 XML 文档中出现二者之一。另外用个内容模式节点

(!ELEMENT facher,sonl,(son2 |son3),son4>

来体现 son2 和 son3 之间的选择关系。元素 father 对应的局部元素树如图 4 一 3。

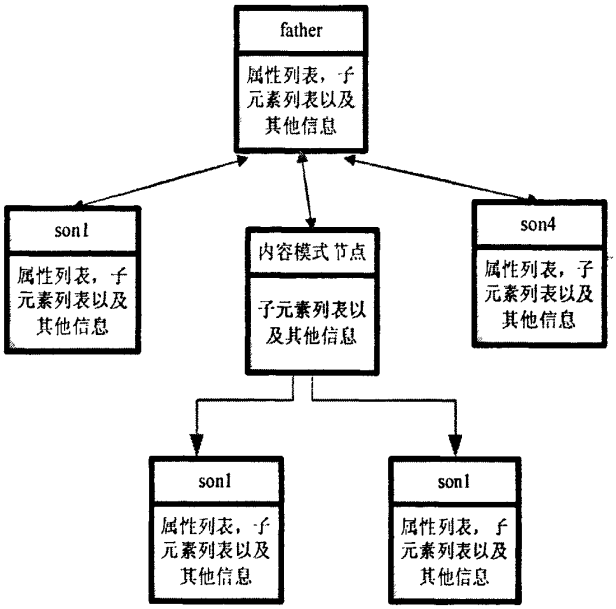


图 4-3 元素树

Fig.4-3 Element tree

元素树的生成算法如下：

1. 解析给定的 DTD，对于所有不包含在其他元素的元素内容中的元索 R，构造一棵以元素 R 为根节点的元素树，

2 创建一个节点，其节点名为 R，

3 根据已解析的 DTD，获取元素 R 的属性列表 Attributes，并将 Attributes 加入到节点 R 中。

4 分析元素 R 的内容模式，获取节点 1{的其他信息，并根据每一个元素内容的类型生成节点 R 的子元素列表 EL：

4.1 如果 EL 是 EMPrY 类型，则该节点不包青子元素列表，是一个叶节点。

4.2 如果 EL 是混合类型．则为其中的字符数据，生成一个节点名为 PCDATA 盯 A 的叫节点；对于其他内存继续执行 5。

4.3 如果 EL 是元素内容类型. 则继续执行 5。

5 根据内容模式里每一个内容粒予 cP 的类型，生成节点 F 的了元素列表：

5. 1 如果 CP 足名字 s，则生成子节点 S，然后找到元素 S，并执行 3，4，5 生成以 S 为根的于树。

5. 2 如果 CP 是内容粒子序列列表或选择列表，则对于列表中的每一个子内容粒子 SubCP：

5.2.1 如果 SubCP 是一个名字. 则执行 5. 1。

5.2.2 如果 SubCP 是个序列列表或选择列表，则生成一个内容模式节点作为 R 的子节点，然后执行 5 2，生成以该内容模式节点为根的子 树。

4.3.2 映射关系

在元素树的基础上，我们还需要定义 XML 元素与其他类型数据之间的映射关系。对于结构化数据，由于其具有良好的结构，我们可以先定义 XML 文档与该结构化数据在结构上的映射规则.然后利用映射规则自动确定 XML 元素与业务数据的映射关系。

以数据库为例，元素树与数据库模式之间的映射规则可以定义如下：

●元素树的根节点对应数据库中的一个表，称为根表；

●如果一个节点的属性列表或子元素列表不为空，则该节点称为表节点，它对应数据库中的一个表：

●一个表节点的每个属性和子节点都对应于该节点所对应的表中的一个字段；

●如果一个表节点有父节点，那么父节点所对应的表称为该节点所对应的表的父表.

●数据库中，除了根表以外的表都应该包含一个与其父表发生关联的外键；

●除了表节点和内容模式节点以外的节点称为字段结点，它只对应父节点所对应的表中的一个字段。

●内容模式节点小对应数据库巾的任何对象。

然而在实际应用中用户数据的结构(如数据库模式)往往已经建立好

了，这时我们需要由用户指定 XML 元素与业务数据之间的映射关系，并且将这些数据保存到一个映射表。对于非结构化数据，我们同样需要设计一个映射表，用来保存 XML 元素与业务数据的映射关系。

4.3.3 数据转换

基于元素树的转换方法在元素树的基础上，根据映射关系制定一系列的执行指令。通过执行这些指令．并将执行结果插入到数据模型中的相应位置，就可得到相应 XML 文档。同样，执行反向指令就可以把 XML_文档转换为其他格式的数据。由于引入了内容模式节点，系统对 XML 文档结构的限制大大放宽了。而且由于元素树的生成是基于 DTD 的，因此对于符合同一个 DTD 的类 XML 文档的转换，该 DTD 所对应的元素树可以被多次复用。在企业电子商务中，企业需要交互的 XML 文档对应的 DTD 相对来说是固定的。一旦所有 DTD 的元素树都生成了，系统可以复用已有的元素树，从而大大提高系统的性能[24]。

4.4 本章小结

基于 XML 文档的数据转换有两种常用的方法：基于模板的数据转换方法和基于模型的数据转换方法。前者在进行转换前并不事先定义好 XML 文档与其他数据之间的映射关系，而是在 XML 文档中嵌入一些可执行的指令。通过指令的执行，将其他格式的数据插入到 XML 模板中，从而生成 XML 文档，该方法只能完成单向的转换。后者在进行转换前先建立一个数据模型，该模型体现了 XML 文档与其他数据之间的映射关系。通过对模型的操作，实现 XML 文档与其他格式数据的双向转换。基于元素树的数据转换方式是一种基于模型的转换方法。该方法首先创建元素树，然后定义元素树节点与其他类型数据之间的映射关系，并在元素材和映射关系的基础上完成数据的转换工作

第五章 数据转换系统的分析

5.1 数据转换系统模型

异构数据库数据交换是由多个异构的成员数据库组成的数据库系统集合，以 XML 作为中间件，实现数据的交换、共享和透明访问。每个数据库在加入异构数据库系统之前，都是独立存在的，并且拥有完整的 DBMS，在实现数据库数据交换之前，每个数据库都保持自身的完整性和安全性，需要交换的两个或多个数据库系统，其数据表和结构是相似和相同的。异构数据库是指包含不同物理模式、不同数据模型的数据库，同数据模型不同厂商的同质异型数据库，以及同一数据库厂商的不同版本、针对不同网络环境的数据库产品等等为了实现异构数据库之间的数据传送，就一定要首先找出它们之间的差异。引起数据库差异的因素很多，如：计算机硬件、操作系统、数据模型、物理模型、数据语义等的不同。异构数据库数据交换系统的异构特性主要表现在以下方面[25]：

1. 数据库系统本身的异构：可以是不同数据模型的数据库，如层次数据库、网状数据库，还有现在非常流行的关系型数据库，以及将来大力推广的对象型数据库。也可以是不同生产厂商生产的关系型数据库，如 MYSQI.,、MSSQI.、ORACLE 等。

2.操作系统的异构：使用数据库的大型应用程序可以运行在 Windows、UNIX、Linux 等操作系统上，这些操作系统运行机制和内核完全不 同。

3.计算机体系结构的异构：各个参与的数据库可以运行在并行机、大型机、服务器、工作站、PC 机和 PDA 上。

4.应用程序的异构：各个参与的数据库可以运行在基于 C / S 和 B / S 结构的应用系统上。在基于 XML，和 Web Services 的异构数据库数据交换系统的研究与设计过程中，面临的主要难题就是，在一个应用中或系统中，如何有效的实现两个或多个数据库的连接、数据交换、数据共享和数据的透明访问。对于异构数据库基于 XMI，和 Web Services 异构数据转换的设

计与实现系统，实现数据共享应该达到两点：一是数据库的数据转换，二是实现数据的透明访问。其中数据库的数据转换还包括源数据库与目的数据库的结构模式的映射，在数据库结构映射完成之后，再实现数据库中数据的映射。

在异构数据系统中实现了数据的透明访问，用户就可以将异构分布式数据库系统看成普通的分布式数据库系统，用自己熟悉的数据处理语言去访问数据库，如同访问一个数据库系统一样。但目前还没有一种广泛使用的数据定义模型和数据查询语言，实现数据的透明访问可以采用多对一转换、双向的中间件等技术。开放式数据库互连(Open DataBase Connectivity，简称 ODBC)是一种用来在相关或不相关的数据库管理系统中存取数据的标准应用程序接口(API)。ODBC 为应用程序提供了一套高层调用接口规范和基于动态链接库的运行支持环境。ADO(ActiveX Data Object)即 ActiveX 数据对象，是一组优化的访问数据库的专用对象集，它为 ASP 提供了完整的站点数据库解决方案，它作用在服务器端，提供含有数据库信息的主页内容，通过执行 SQL 命令，让用户在浏览器画面中输入，更新和删除站点数据库的信息。JDBC(Java DatabaseConnectivity)是支持基本 SQL 功能的一个通用的应用程序编程接口，它在不同的数据库功能模块的层次上提供了一个统一的用户界面，为对异构数据库进行直接的 Web 访问提供了新的解决方案。JDBC 已被越来越多的数据库厂商、连接厂商、Internet 服务厂商及应用程序编制者所支持。

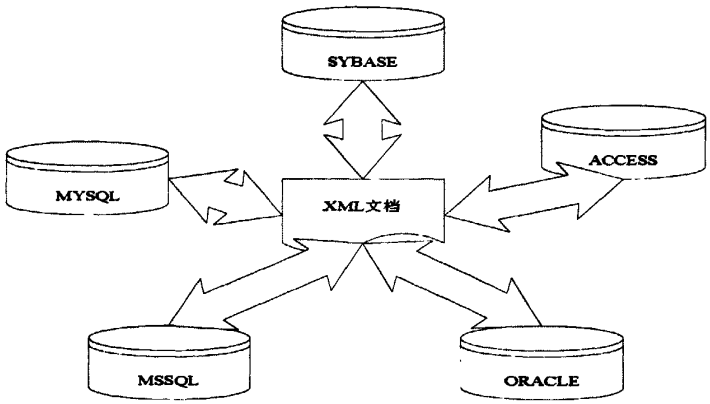


图 5-1 数据库数据交换模型

Fig.5-1 Database data exchange model

XML 是一种结构化的数据描述语言，具有简单、自描述性、与平台无

关性、能够自定义标记等特性，鉴于上述优点，XML 文档非常适合作为异构数据库数据交换的中间件。基于 XML 的异构数据库数据交换系统，是以 XML 作为中间件来实现异构数据库数据交换的模式转换和数据转换。首先，XML，的与平台无关性，满足异构数据库的所需要的跨平台性；XML。的自描述性和结构化特性能够非常清晰的获取数据库的元数据信息；XML，的可格式化，可以使用 XSLT 将 XML 文档描述成各种表现形式如 HTML、XHTML，等。目前，XML 技术已经逐渐开始应用于异构数据库的数据交换，并且已成为事实上的数据交换标准，以 XML 为公共数据模型转换异构关系数据库也为关系数据库与其他数据类型的集成转换提供了便利。

5.2 提取用于数据交换的 XML 标准文档

通常情况下我们在编写 XML 文档时用到什么标记就创建什么标记，没有对这种标记和标记间依赖关系进行定义。这样作为文档的创作者可能会非常清楚标记的意义和标记问的依赖关系，但作为文档的使用者或处理该文档的程序而言就不明其义了。对于数据交换来说，最重要的是数据交换的双方应有一个统一的数据格式，只有采用统一的数据格式，才能实现数据的自动流转、处理等功能。其中每一个系统都将其内部的数据转换成行业标准的基于 XML 的数据格式，这样就可以实现不同系统之间的数据交换，如果有行业标准的基于 XML 的数据格式，则可以直接采用之；如果没有行业标准，则编制企业内部标准．并对外界公布，我们称之为接口，外界通过这些接口．实现与系统的交互。例如，我们定义基于 XML 的查询模板，用户根据这个模板接口编写特定的查询文件(XML 文档)并发送至系统，系统数据转换构件接收该文件，提取出数据，转换为 XML 文件返回给用户。

在实际应用中，XML 文档没有结构描述部分，但它自身的层次关系就可以体现 XML 文档内数据的结构，因而能够清晰地表达数据之间的依赖关系。所以要充分利用 XML 文档的层次结构和标识，这两种方法可以清晰地表示 XML 文档内部对象之间的关系。

提取用于数据交换的 XML 标准文档，就是生成数据交换文件的 DTD(或 XML Schemal)，它对数据交换文件(XML 文档)的合法性检验。

5.3 关系模式与 XML 模式相互转换脚本

要在各种电子商务，电子政务之间交换数据，必须先在 XML 文档和数据库之间转换数据，把 XML 模式映射到数据库模式和把数据库模式映射到 XML 模式。

解决 XML 模式和数据库模式相互映射问题，有两种选经：基于表格的映射和基于关系对象的映射(第四章)。这里采用基于表格的映射来解决 XML 模式和数据库模式相互映射问题。对于每个具体的关系模式与 XML 模式之间的相互转换，都要编写一个转换脚本，对于具体某个关系数据库，依据表单 DTD(或 XML Schema)，定义从关系模式到 XML 模式的转换脚本，和从 XML 模式到关系模式的转换脚本。转换方法示意图如图 5—2：

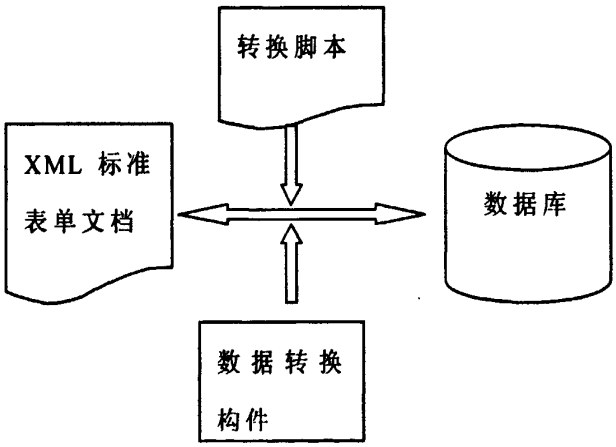


图 5-2 XML 与数据库转换图

Fig.5-2 XML and Database Conversion

在转换脚本中嵌入预定义的查询语句(基于模板的查询)，由数据转换构件执行查询语句得到表格式的数据，然后再使用转换脚率的转抉规则将表格式的数据转换为结构化的 XML 文档数据。

5.4 基于 XML 的数据转换构件

在定义了数据转换脚本之后，数据转换构件使用这些转换脚本在关系数据库和 XML 文档之间相互转换数据。

在数据交换中使用表单 XML 文档的问题并不是简单地共享一个表单模式，在贸易合作伙伴间按照模式发送表单。贸易合作伙伴间的数据库是异构的：发送的表单是多样的，因此有必要定义一个关系模式与一种表单 XML 文档之间的转换脚本，并调用数据转换构件的相应接口，通过深度遍历转换脚本，逐个分支拓展创建，最终完成整个 XML，文档的 DOM 树的创建，实现广度的实体关系到深度的实体关系的映射，能够从关系数据库中提取出发送给任何贸易合作伙伴的表单 XML 文档。通过使用转换脚本能实现发送或接收的单据模式本地化。

本系统的主要构件是数据库接口构件和数据转换构件。

数据库接口构件的功能是用以连接各种类型的数据库。

数据转换构件的功能是：依照数据发送方(或接收方)的转换脚本实现关系数据库与 XML 文档之间的转换。

数据转换构件的工作内容包括：1)从关系数据库中提取出表格形式的数据集，依照表单转换规则脚本 DOM 树的结构，将数据集转换为 XML 文档。如图 5—3 所示；

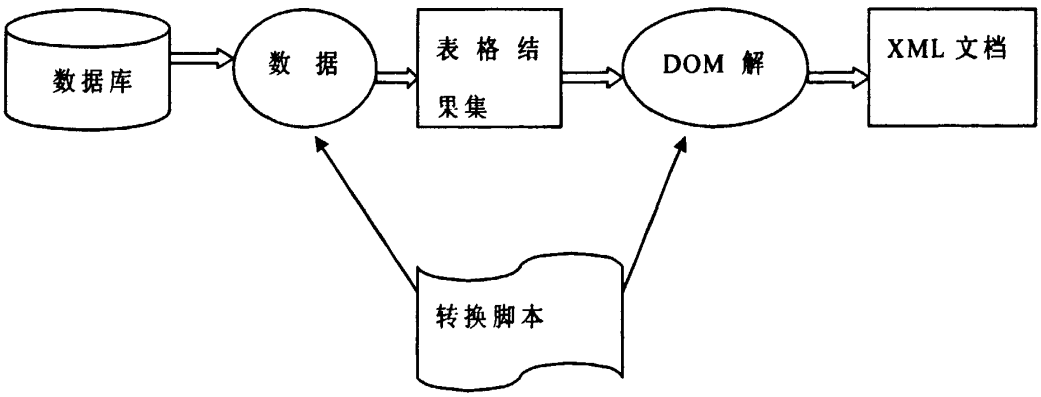


图 5-3 从数据库到 XML 文档的数据转换

Fig.5-3 XML documents from databases to data conversion

2)利用 DOM 解析工具分析法送来的表单。XML 文档，并根据 DTD(或 XML Schema)判断其有效性，执行嵌入表单 XML 文档中的转换脚本命令。转换 XML 文档，将转换后的数据存入关系数据库中。如图 5_4 所示：

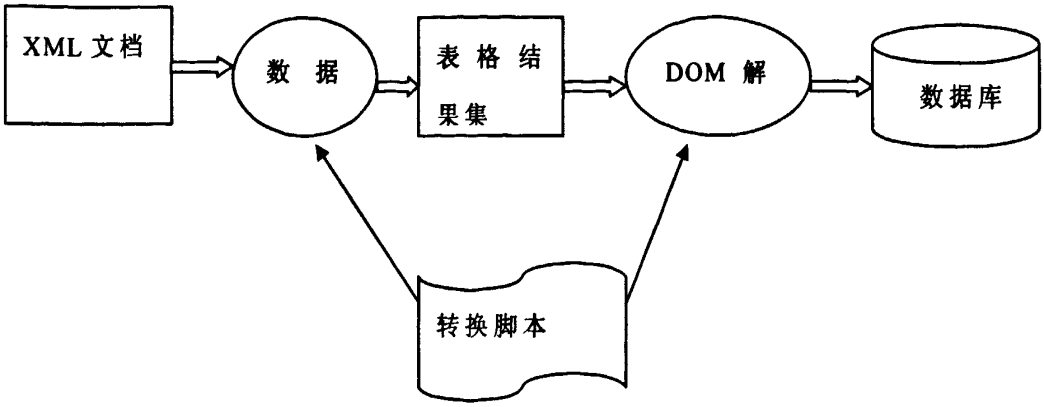


图 5-4 从 XML 文档到数据库的数据转换

Fig.5-4 XML documents from data conversion to the database

数据转换系统使用 DTD 文件、转换脚本和构件相结合的结构，可以提高数据转换系统的灵活性和可维护性。随着系统的演进，数据交换的内容可能会发生变化，例如增加数据的字段，此时，只修改 DTD 文件和转换脚本，不必修改构件的代码，而 DTD 文件和转换脚本是文本文档，容易修改。因此，系统可维护性很好，在实际应用中也体现了这一点。

5.5 本章小结

本章通过对系统要实现的功能的需求分析提出了数据转换系统模型，依据上章选择的基于模型的数据转换方法，介绍了其中所涉及的用于数据交换的 XML 文档，以及关系模式与 XML 模式的相互转换脚本。

第六章 系统的设计与实现

数据转换系统包含 DTD（XML Schema）文件、转换脚本、转换构件和数据库接口构件以下以全国农业网站数据交换系统中的实例说明它们的设计与实现。

目前国内各家农业信息网站信息类别和网站栏目大同小异，没有一个统一的数据存储格式，并且采用了不同的数据库平台和结构，自成体系，为相互之间实现数据的交换和共享造成了障碍，为此亟待制定一种数据交换标准来消除这一障碍。随着近两年XML标准的出现和日益成熟，并很快成为各种复杂的异构数据的交换得以实现的核心技术。由于XML是非专有的并易于阅读和编写，就使得它成为在不同的应用间交换数据的理想格式。制定此基于XML格式的数据转换格式规范主要是立足于我国各家农业网站的需要，通过广泛借鉴现有的国内外相关标准，应用XML及其相关技术，建立完善的原型体系，通过试点测试，制定一套科学实用的《全国农业网站数据交换格式规范》。根据此规范和应用格式转换软件，可实现数据格式和内容的分离；利用元数据规范定义元素的结构，并根据元素的使用和要求来对信息资源进行描述和分类，使得寻找和使用信息资源的过程更快捷有效，从而方便各家农业网站之间信息进行自由交换和共享，最终提高信息访问率和网站运行效率，大大提高了社会效益。

在分析我国农业网站栏目的结构和特点的基础上，我们选择了网站资讯信息、供求信息、价格信息、农业科技四个栏目目标制定格式交换标准，网站其他栏目可以参照这些栏目进行交换格式定义。

6.1 数据交换流程

图6-1中模型描述了网站数据从采集到交换的一般过程。各网站在发布数据交换文档的时候需提供相应的Schame且通过XML的合法性校验。

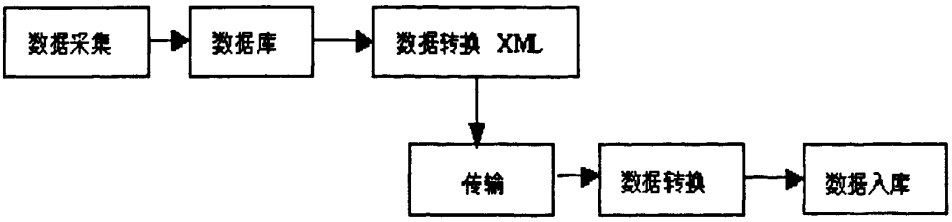


图 6-1 网站数据采集、交换过程

Fig.6-1 data collection and exchange process

6.2 网站信息的基本要素

基于 XML 的网站信息应能满足网站内各类数据交换的需要，组成要素应满足网站上各栏目信息的需要。基本要素包括如下：

——数据交换信息（提供单位、时间）

——数据分类栏目信息

——各栏目交换的数据条目

各个要素层次关系如图6-2所示：

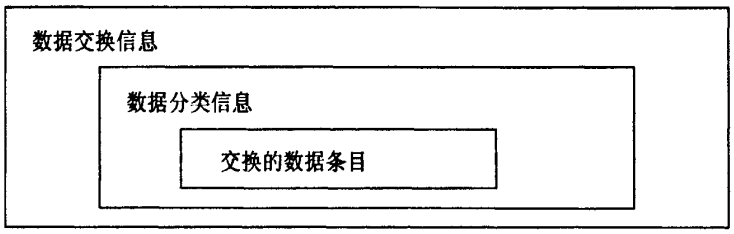


图 6-2 网站信息基本要素

Fig.6-2 Basic elements of information

6.3 系统的数据结构模型

6.3.1 数据交换

整个数据交换体由发布单位、发布时间、说明以及栏目组成。其结构模型如图6-3所示。

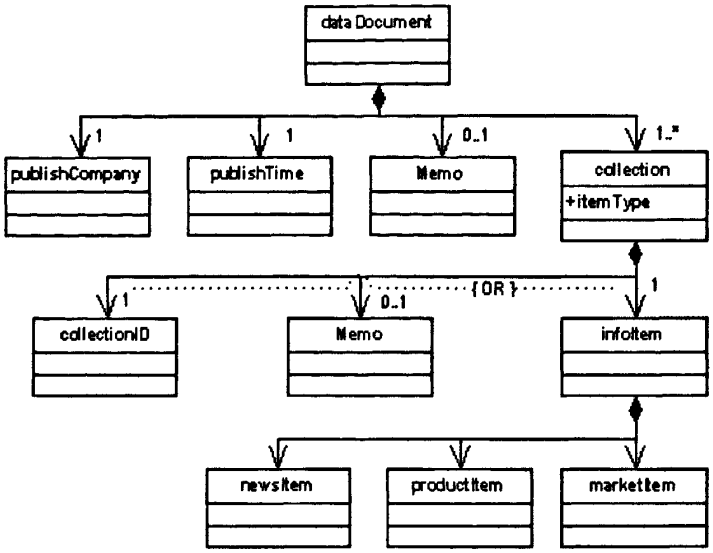


图 6-3 数据交换体 UML 结构模型

Fig.6-3Data Exchanger UML Model

6.3.2 各栏目的结构模型

由于网站各个栏目信息结构，信息内容组成要素有很大区别，需要对各个栏目的信息条目组成设置不同的格式。

资讯类信息条目的UML结构模型见图6-4。

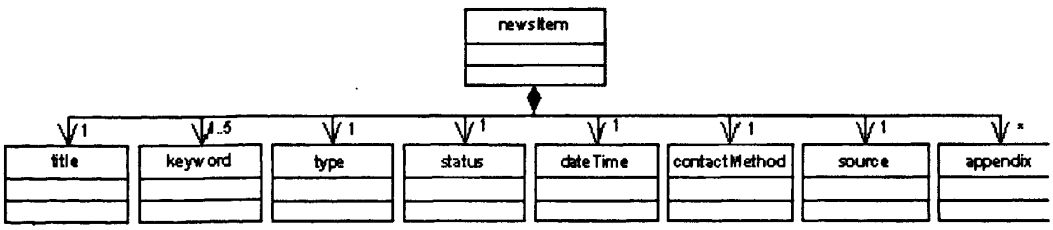


图 6-4 资讯类信息条目的 UML 结构模型

Fig.6-4 Information type information entry UML model

供求信息类信息条目的UML结构模型见图6-5。

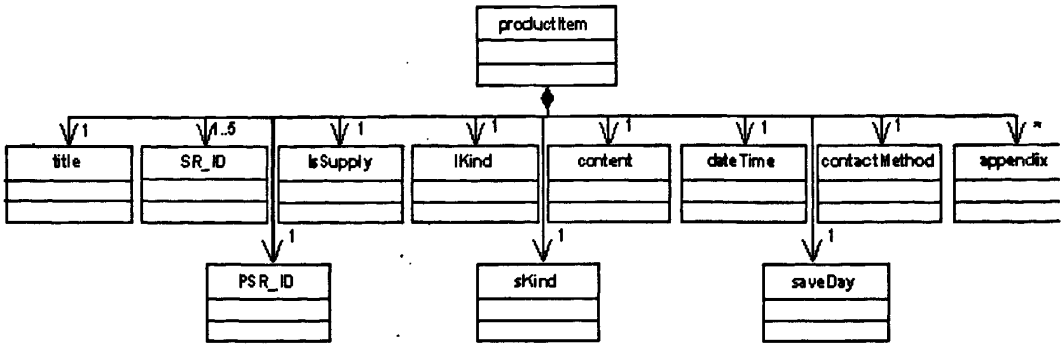


图 6-5 供求信息类信息条目的 UML 结构模型

Fig.6-5 Supply and demand information entry UML class model of information

市场行情类信息条目的UML结构模型见图6-6。

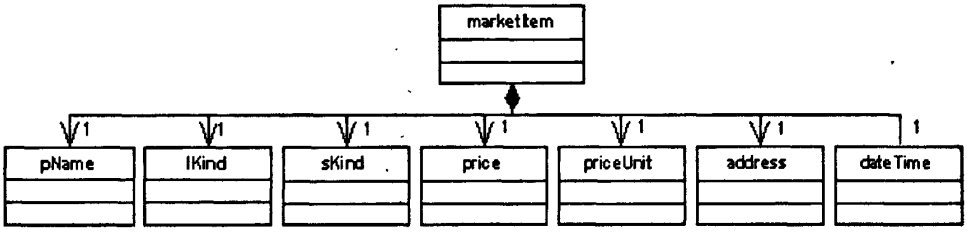


图 6-6 市场行情类信息条目的 UML 结构模型

Fig.6-6 Market entry of the UML class model of information

其中各信息条目中的联系方式的UML结构模型见图6-7。

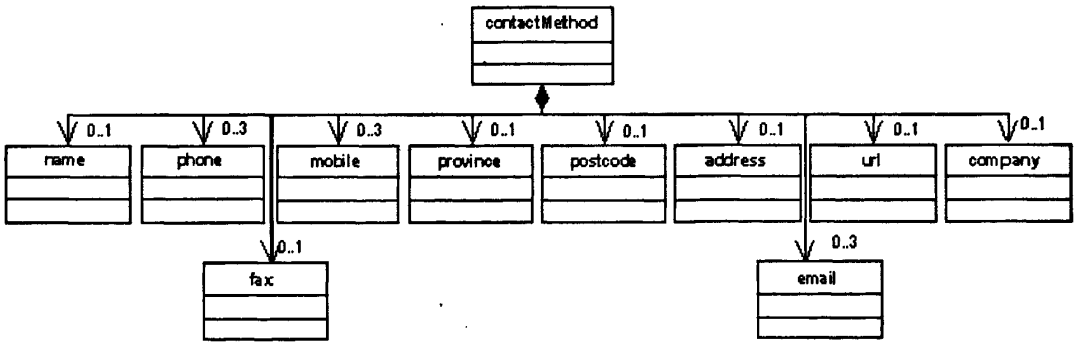


图 6-7 联系方式的 UML 结构模型

Fig.6-7 The UML model of contact

6.4 数据描述

6.4.1 数据交换

XML标记：dataDocument

定义：记录数据交换的各组成要素的有序集合

命名空间为“http://www.ahagri.com/DataExchange”。

值域：不做要求。

DTD定义：

<!ELEMENT dataDocument (publishCompany, publishTime, Memo, collection+)>

<!ATTLIST dataDocument

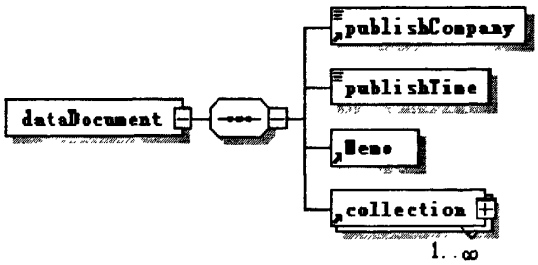
xmlns CDATA #FIXED http://www.ahagri.com/DataExchange

>

Schema定义：见表6-1。

表 6-1 数据交换 Schema 定义

TABLE.6-1 Data Exchange Schema Definition

层次模型	
子元素	publishCompany publishTime Memo collection
命名空间	http://www.ahagri.com/DataExchange
源代码	<xs:element name="dataDocument"> <xs:complexType> <xs:sequence> <xs:element ref="publishCompany"/> <xs:element ref="publishTime" />

源代码	<xs:element ref="Memo"/> <xs:element ref="collection" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element> <xs:element name="dataDocument"> <xs:complexType> <xs:sequence> <xs:element ref="publishCompany"/> <xs:element ref="publishTime" /> <xs:element ref="Memo"/> <xs:element ref="collection" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>

现以其中的“发布单位”为例进行介绍

XML标记：publishCompany

定义：能够标识数据交换文档的发布单位唯一标志。

值域：不作要求。

DTD定义：<!ELEMENT publishCompany (#PCDATA)>

Schema定义：见表6-2。

表 6-2 publishCompany Schema 定义

TABLE.6-2 publishCompany Schema Definition

数据类型	xs:string
父元素	dataDocument
源代码	<xs:element name="publishCompany" type="xs:string"/>

现以其中的“栏目”为例进行介绍

XML标记：collection

定义：记录要交换的各栏目信息和需要交换的数据。

值域：不做要求。

DTD定义：

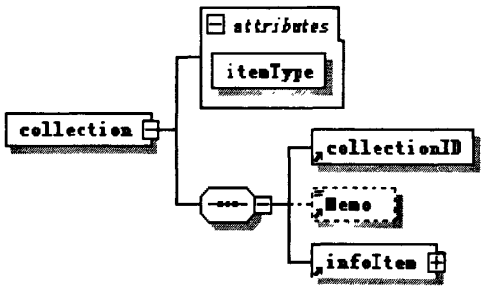
<!ELEMENT collection (collectionID, Memo, infoItem)>

<!ATTLIST collection

CDATA #REQUIRED>

Schema定义：见表6-5。

表 6-5 collection Schema 定义

层次模型	
父元素	dataDocument
属性列表	itemType
子元素	collectionID Memo infoItem
源代码	<xs:element name="collection"> <xs:complexType> <xs:sequence> <xs:element ref="collectionID"/> <xs:element ref="Memo"/> <xs:element ref="infoItem"/> </xs:sequence> <xs:attribute name="itemType" type=" itemType " use="required"/> </xs:complexType> </xs:element>

现以其中的“信息条目”为例进行介绍

XML标记：infoItem

定义：记录栏目中要交换的数据。

值域：不做要求。

DTD定义：

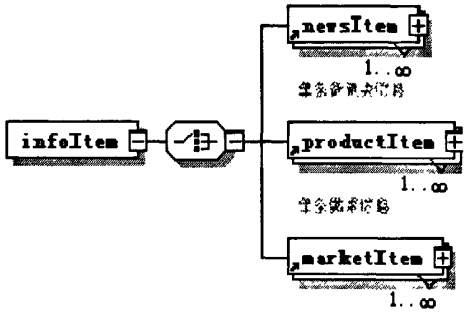
<!ELEMENT infoItem (newsItem+ | productItem+ | marketItem+)>

注：各栏目中信息条目里具有相同性质的信息，包括联系方式、图片等元素定义参考6.2

Schema定义：见表6-8。

表 6-8 infoItem Schema定义

TABLE.6-8 infoItem Schema Definition

层次模型	
子元素	各栏目要交换的的数据条目
父元素	栏目
源代码	<xs:element name="infoItem"> <xs:complexType> <xs:choice> <xs:element ref="newsItem" maxOccurs="unbounded"/> <xs:element ref="productItem" maxOccurs="unbounded"/> <xs:element ref="marketItem" maxOccurs="unbounded"/> </xs:choice> </xs:complexType> </xs:element>

现已其中的“供求信息条目”为例进行介绍

XML标记：productItem

定义：记录供求信息栏目中信息的标题，类型，信息类别，内容，发布日期，有效期限，联系人，联系方式，图片。

值域：不做要求。

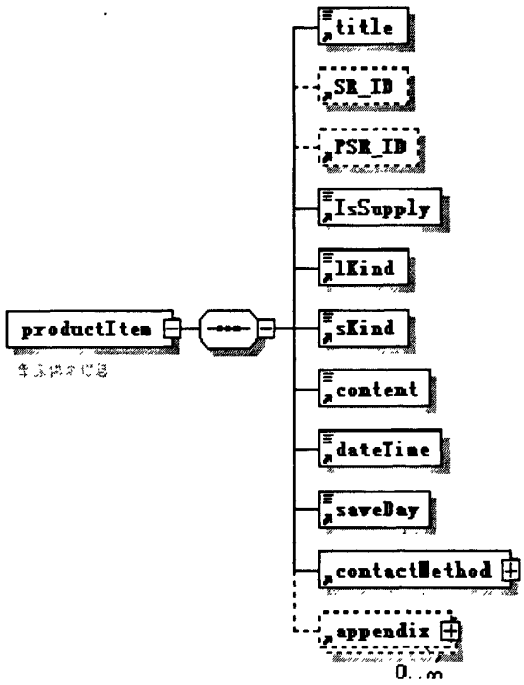
DTD定义：

<!ELEMENT productItem (title, SR_ID?, PSR_ID?, IsSupply, lKind, sKind, content, dateTime, saveDay, contactMethod, appendix*)>

Schema定义：见表6-17。

表 6-17productItem Schema 定义

TABLE.6-17 productItem Schema Definition

层次模型	
子元素	title SR_ID PSR_ID IsSupply lKind sKind content dateTime saveDay contactMethod appendix
父元素	infoItem
源代码	<xs:element name="productItem"> <xs:complexType> <xs:sequence> <xs:element ref="title"/>

	<xs:element ref="SR_ID" minOccurs="0"/> <xs:element ref="PSR_ID" minOccurs="0"/> <xs:element ref="IsSupply"/> <xs:element ref="lKind"/> <xs:element ref="sKind"/> <xs:element ref="content"/> <xs:element ref="dateTime"/> <xs:element ref="saveDay"/> <xs:element ref="contactMethod"/> <xs:element ref="appendix" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> </xs:element>

标题

XML标记：title

定义：供求信息类栏目中信息的标题。

值域：最长20字汉字。

DTD定义：<!ELEMENT title (#PCDATA)>

Schema定义：见表6-18。

表 6-18 title Schema 定义

TABLE.6-18 title Schema Definition

父元素	productItem
源代码	<xs:element name="title" type="limitLen20Type"/>

SR_ID

XML标记：SR_ID

定义：供求信息类栏目中信息的编号。

值域：最长20字汉字。

DTD定义：<!ELEMENT SR_ID (#PCDATA)>

Schema定义：见表6-19。

表 6-19 SR_ID Schema 定义

TABLE.6-19 SR_ID Schema Definition

父元素	productItem
源代码	<xs:element name="SR_ID" type="limitLen20NumType"/>

PSR_ID

XML标记：PSR_ID

定义：供求信息类栏目中信息的父编号，供类型为回复时使用。

值域：最长20字汉字。

DTD定义：<!ELEMENT PSR_ID (#PCDATA)>

Schema定义：见表6-20。

表 20 PSR_ID Schema 定义

TABLE.6-20 PSR_ID Schema Definition

父元素	productItem
源代码	<xs:element name="PSR_ID" type="limitLen20NumType"/>

类型

XML标记：IsSupply

定义：供求信息类栏目中信息的类型。

值域：供应：Y；求购：N；回复 ：R。

DTD定义：<!ELEMENT IsSupply (#PCDATA)>

Schema定义：见表6-21。

表 6-21 productItem Schema 定义

TABLE.6-21 productItem Schema Definition

父元素	productItem
源代码	<xs:element name="IsSupply"> <xs:simpleType> <xs:restriction base="xs:string">

	<xs:enumeration value="Y"/> <xs:enumeration value="N"/> <xs:enumeration value="R"/> </xs:restriction> </xs:simpleType> </xs:element>

大类

XML标记：lKind

定义：供求信息类栏目中信息的大类。

值域：参见附录C.2产品。

DTD定义：<!ELEMENT lKind (#PCDATA)>

Schema定义：见表6-22。

表 6-22 lKind Schema 定义

TABLE.6-22 lKind Schema Definition

父元素	productItem
源代码	<xs:element name="lKind" type="limit2Num"/>

小类

XML标记：sKind

定义：供求信息类栏目中信息的小类。

值域：参见附录C.2产品分类表。

DTD定义：<!ELEMENT sKind (#PCDATA)>

Schema定义：见表6-23。

表 6-23 sKind Schema 定义

TABLE.6-23 skind Schema Definition

父元素	productItem
源代码	<xs:element name=" sKind " type="limit2Num"/>

内容

XML标记：content

定义：供求信息类栏目中信息的内容。

值域：不做要求。

DTD定义：<!ELEMENT content (#PCDATA)>

Schema定义：见表6-24。

表 6-24 content Schema 定义

TABLE.6-24 content Schema Definition

父元素	productItem
源代码	<xs:element name="content" type="xs:string"/>

发布时间

XML标记：dateTime

定义：供求信息类栏目中信息的发布时间。

值域：不做要求。

DTD定义：<!ELEMENT dateTime (#PCDATA)>

Schema定义：见表6-25。

表 6-25 dateTime Schema 定义

TABLE.6-25 dateTime Schema Definition

父元素	productItem
源代码	<xs:element name="dateTime" type="xs:dateTime"/>

有效期限

XML标记：saveDay

定义：供求信息类栏目中信息的有效期限。

值域：不做要求。

DTD定义：<!ELEMENT saveDay (#PCDATA)>

Schema定义：见表6-26。

表 6-26 saveDay Schema 定义

TABLE.6-26 saveDay Schema Definition

父元素	productItem
源代码	<xs:element name="saveDay" type="xs:positiveInteger"/>

6.4.2 信息数据类型定义

现已“图片”为例进行介绍

XML标记：appendix

定义：记录信息条目中的图片信息。

值域：不作要求。

DTD定义：

```
<!ELEMENT appendix (#PCDATA)>
<!ATTLIST appendix
  appFileName CDATA #REQUIRED
  appFileExt CDATA #REQUIRED
  appMemo CDATA  #REQUIRED>
```

Schema定义：见表6-35。

表 6-35 appendix 的 Schema 定义

TABLE.6-35 appendix Schema Definition

父元素	newsItem productItem marketItem
属 性	appFileName appFileExt appMemo
源代码	<xs:element name="appendix"> <xs:complexType> <xs:attribute name="appFileName" use="required"/> <xs:attribute name="appFileExt" use="required"/> <xs:attribute name="appMemo" type="limitLen20Type" use="required"/> </xs:complexType> </xs:element>

现以联系方式进行介绍

XML标记：contactMethod

定义：记录各信息条目中的联系方式，如姓名、电话号码、传真号码、手机号码、省份、邮编、地址、单位、电子邮件、网址等，从信息条目中提取出来单独描述。

值域：不做要求。

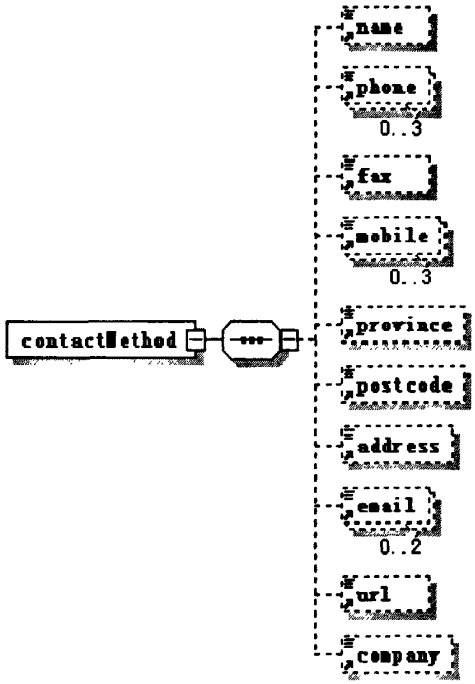
DTD定义：

<!ELEMENT contactMethod (name?, phone*, fax?, mobile*, province?, postcode?, address?, email*, url?, company?)>

Schema定义：见表6-36。

表 6-36 contactMethod 的 Schema 定义

TABLE.6-36 contactMethod Schema Definition

层次模型	
父元素	newsItem productItem marketItem
子元素	Name phone fax mobile province postcode address email url company
源代码	<xs:element name="contactMethod"> <xs:complexType> <xs:sequence> <xs:element ref="name" minOccurs="0"/>

	<xs:element ref="phone" minOccurs="0" maxOccurs="3"/> <xs:element ref="fax" minOccurs="0"/> <xs:element ref="mobile" minOccurs="0" maxOccurs="3"/> <xs:element ref="province" minOccurs="0"/> <xs:element ref="postcode" minOccurs="0"/> <xs:element ref="address" minOccurs="0"/> <xs:element ref="email" minOccurs="0" maxOccurs="2"/> <xs:element ref="url" minOccurs="0"/> <xs:element ref="company" minOccurs="0"/> </xs:sequence> </xs:complexType> </xs:element>

现已其中的“联系人姓名”进行介绍

XML标记：name

定义：联系人的姓名。

值域：不做要求。

DTD定义：<!ELEMENT name (#PCDATA)>

Schema定义：见表6-37。

表 6-37 name Schema 定义

TABLE.6-37 name Schema Definition

父元素	contactMethod
源代码	<xs:element name="name" type="limitLen10Type"/>

6.5 关键算法的实现

数据转换的关键算法的实现：

1.从数据库到 XML 文档的转换 ConverToXML(scriptfile. xmlfile)

具体的实现过程为建立 DOM 解析器，解析脚本文件 scriptfile，取出根节点 scriptroot，建立空 XML 文档到 xmlDoc，依照脚本文件中处理指示建立 xmlDoc 的处理指示，将脚本文件根节点及其属性添加到 xmlDoc 中，如果根节点有 sql 属性，直接调用 GenerateTreeFromDB(scriptRoot，xmlRoot). 再则调用函数 ParseConverToxml(scriptRoot. xmlRoot)分析脚本文件并依据其中的描述从数据库中循环读记录，转换后添加到 xmlRoot 节点下。最后，将 xml 文档 xmlDoc 输出到结果文件 xmlfile 中。

2 从 XML 文档到数据库的转换 ConverToDB(scriptfile，xmllfile)

具体的实现过程为建立 DOM 解析器，解析脚本文件 scriptfile，取出根节点 scriptRoot，解析输入的 XML 文档 xmlfile，取出其根节点 sourexmlRoot，如果脚本文件的根节点有 sql 属性，直接调用 ParseRecSQLScriptTree() 和 GenerateDBFromTree()，否则调用 ParseConverToDB(sourexmlRoot，curTable)，建立空 xml 文档 xmlDoc，依照脚本文件中处理指示建立 xmlDoc 的处理指示。最后，依据输入 xml 文档中的处理指令，稚撅据库中将 xmlDoc 的数据进行插入、更新或删除操作。

6.6 系统实现

根据对我国的各家农业信息网站分析后可得出以下的栏目、产品分类表

1 栏目分类表

表6-47 栏目分类表

Part classification

分类ID	对应频道	信息说明
72	灾情预测	全国各省农业灾情信息
77	气象新闻	全国各省重要气象天气信息
124	西部新闻	与西部有关的农业信息
141	民族技艺	地方民族风情
142	小吃特产	地方名吃特产
143	名胜古迹	地方旅游景点、名胜古迹
171	网上农展	全国和各省举办各类农业展会信息
173	招商引资	全国和各省的招商引资信息
214	省级快讯	省市内与“三农”有关的重要信息
215	领导关怀	领导对农业工作的支持与关注信息
310	科技新闻	与农业相关的科技新闻
320	实用技术	农业实用技术
608	供求信息	供求信息
618	市场行情	市场行情

2 产品分类表

表6-48 产品分类表

Product classification

大类名称	大类 ID	小类名称	小类 ID
农副产品	01	粮食	01
		油料	02
		棉麻烟	03
		蔬菜	04
		水果	05
		水产品	06
		茶叶	07

		副食	08
		其它	99
农业物资	02	农药	01
		化肥	02
		种子	03
		饲料	04
		农机	05
		其它	99
林木	03	原材	01
		种苗	02
		花卉	03
		制品	04
		机械	05
		其他	99
药材	04	畜药	01
		禽药	02
		鱼药	03
		中药材	04
		其他	99
农机器械	05	农机	01
		器械	02
		其他	99

根据前面章节的分析可以得出有关农业网站的数据交换 DTD 定义以及数据交换的 Schema 定义：

1 数据交换的 DTD 定义形式如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<!ELEMENT dataDocument (publishCompany, publishTime, Memo,
collection+)>
<!ATTLIST collection
```

```
CDATA #REQUIRED
>
<!ELEMENT infoItem (newsItem+ | productItem+ | marketItem+)>
<!ELEMENT collection (collectionID, Memo?, infoItem)>
<!ELEMENT newsItem (title, (keyword, keyword, keyword?, keyword?,
keyword?, keyword?, keyword?), type, status, dateTime, (contactMethod,
contactMethod, contactMethod?, contactMethod?, contactMethod?), source?,
appendix*)>
<!ELEMENT productItem (title, SR_ID?, PSR_ID?, IsSupply, lKind, sKind,
content, dateTime, saveDay, contactMethod, appendix*)>
<!ELEMENT marketItem (pName, lKind, sKind, price, priceUnit, address,
dateTime)>
<!ELEMENT contactMethod (name?, phone*, fax?, mobile*, province?,
postcode?, address?, email*, url?, company?)>
<!ATTLIST appendix
CDATA #REQUIRED
CDATA #REQUIRED
CDATA #REQUIRED
>
<!ELEMENT Memo (#PCDATA)>
<!ELEMENT publishCompany (#PCDATA)>
<!ELEMENT publishTime (#PCDATA)>
<!ELEMENT collectionID (#PCDATA)>
<!ELEMENT PSR_ID (#PCDATA)>
<!ELEMENT IsSupply (#PCDATA)>
<!ELEMENT name (#PCDATA)>
```

2 数据交换的 Schema 定义的形式如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:agriXML="http://www.ahagri.com/agriXML"
```

```
elementFormDefault="qualified"
attributeFormDefault="unqualified">
   <xs:element name="dataDocument">
      <xs:complexType>
         <xs:sequence>
            <xs:element ref="publishCompany"/>
            <xs:element ref="publishTime"/>
            <xs:element ref="Memo"/>
            <xs:element ref="collection" maxOccurs="unbounded"/>
         </xs:sequence>
      </xs:complexType>
   </xs:element>
</xs:schema>
```

3.数据转换系统的XML示例如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<dataDocument xmlns:agriXML="http://www.ahagri.com/agriXML"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="信息交换.xsd">
   <publishCompany>安徽农网</publishCompany>
   <publishTime>2001-12-17T09:30:47.0Z</publishTime>
   <Memo>文档说明</Memo>
   <collection itemType="newsItem">
      <collectionID>115</collectionID>
      <Memo>新闻栏目</Memo>
      <infoItem>
         <newsItem>
            <title>三类人员将成为工会的重点维权对象</title>
            <keyword>维权</keyword>
            <type>10</type>
            <status>1</status>
```

```
            <content>三类人员将成为工会的重点维权对象</content>
            <dateTime>2001-12-17T09:30:47.0Z</dateTime>
            <contactMethod>
               <name>陈玉明</name>
               <phone>01012345678</phone>
               <fax>0101234567</fax>
               <mobile>1300551123</mobile>
               <province>北京</province>
               <postcode>123456</postcode>
   <address>新华网</address>
               <email>sample@ahagri.com</email>
               <url></url>
               <company>String</company>
            </contactMethod>
            <source/>
            <appendix appFileExt="Text" appFileName="Text"
appMemo="String"/>
         </newsItem>
      </infoItem>
   </collection>
   <collection itemType="newsItem">
      <collectionID>214</collectionID>
      <Memo>农业科技栏目</Memo>
      <infoItem>
         <newsItem>
            <title>河蟹成蟹人工养殖及病害防治技术</title>
            <keyword>河蟹 </keyword>
            <type>10</type>
            <status>1</status>
            <content>河蟹成蟹人工养殖及病害防治技术...</content>
```

```
            <dateTime>2001-12-17T09:30:47.0Z</dateTime>
            <contactMethod>
               <name>黄散</name>
               <phone>01012345678</phone>
               <fax>0101234567</fax>
               <mobile>1300551123</mobile>
               <province>北京</province>
               <postcode>123456</postcode>
               <email>sample@ahagri.com</email>
            </contactMethod>
            <source/>
            <appendix appFileExt="jpg" appFileName="picture.jpg"
appMemo="jpg图片"/>
         </newsItem>
      </infoItem>
   </collection>
   <collection itemType="productItem">
      <collectionID>608</collectionID>
      <Memo>供求信息栏目</Memo>
      <infoItem>
         <productItem>
            <title>供优质粉状玉米蛋白粉</title>
            <IsSupply>Y</IsSupply>
<lKind>01</lKind>
            <sKind>01</sKind>
            <content>供优质粉状玉米蛋白粉</content>
            <dateTime>2001-12-17T09:30:47.0Z</dateTime>
            <saveDay>90</saveDay>
            <contactMethod>
               <name>田经理</name>
```

```
                <phone>6332198</phone>
                <fax>05436333315</fax>
                <mobile>1300551123</mobile>
                <province>北京</province>
                <postcode>123456</postcode>
                <address>滨州市无棣院前街</address>
                <email>futian@chinafutian.net</email>
                <url/>
            </contactMethod>
            <appendix appFileExt="Text" appFileName="Text"
appMemo="String"/>
         </productItem>
      </infoItem>
   </collection>
   <collection itemType="marketItem">
      <collectionID>618</collectionID>
      <Memo>市场行情栏目</Memo>
      <infoItem>
         <marketItem>
            <pName>冬瓜</pName>
            <lKind>01</lKind>
            <sKind>04</sKind>
            <price>2.00</price>
            <priceUnit>元/公斤</priceUnit>
            <address>安徽郎溪县城关农贸市场</address>
            <dateTime>2001-12-17T09:30:47.0Z</dateTime>
         </marketItem>
      </infoItem>
   </collection>
</dataDocument>
```

6.7 本章小结

本章以全国农业网站数据的交换为例，详细介绍了数据转换系统的实现环节。通过制定详细的基于 XML 格式的数据转换格式规范和提取标准表单 XML 文档，以及 XML 文档和关系数据库之间数据转换的转换脚本，实现了网站所要求的数据转换系统

结论与展望

随着 Internet 技术和通信技术的快速发展，政务办公系统和电子商务系统在政府和企业中广泛应用，基于现代信息技术和通信技术的“电子政府”应运而生。新的政务办公系统替代旧的办公系统之后，由于系统升级，新旧两套应用系统可能采用不同的数据库作为数据存储源，如何将旧的应用系统中的宝贵的数据资源转换到新的应用系统中，以及如何有效解决各个应用系统中“信息孤岛”的问题。为此，本文提出了基于 XML 的分布式数据交换平台的模型，以 XML 为中间件，实现数据库和 XML 文档之间的双向转换。对于数据库和 XML 文档之间的转换，本文提出了两套转换规则，分别处理从数据库到 XML 文档的转换和从 XML 文档到数据库的转换，从关系模式到 XML 模式的转换相对简单一些，本文提出的转换规则为数据库结构和 XML 文档结构之间建立映射，然后进行数据的转换。从 XML 模式到关系模式的转换相对复杂一些，首先根据 XML 模式自动生成关系模式，然后根据生成的关系模式解析 XML 数据文档，并将数据文档中的数据存储到关系模式中。

数据转换是一项极为重要的数据库操作技术，它关系到应用系统二次开发能力和可移植性。数据转换可以很好的解决信息技术的发展、数据库的升级和分布式技术的发展带来的问题，能够最大限度地利用现有资源，避免重复开发的浪费。

异构数据库的联合使用不仅在数据库原有的应用领域发挥着重要作用，而且被认为在未来新的应用领域也有重要影响。主要领域有：地理信息心痛 GIS、电子商务、电子政务、电子刊物、协同设计。以异构数据库为代表的异构信息源的集成和互用在数据库未来的计算机应用领域都起着关键性的作用。

随着 XML、Java、中间件等相关技术的发展，基于 XML 的应用会越来越多，本课题所做的工作也会不断改进发展。

参考文献

[1]廉迎战，“广东农村信息服务体系模式初探” [J].科技日报，2007，12，23

[2]李海峰．数据库到 xml 文档转换方法[J]．电脑知识与技术，2007（2）

[3]费丽娟，李芸．xml 与关系数据库之间的转换[J]．科技情报开发与经济，2007(21)

[4]李茉莉，基于 xml 的数据库访问技术在图书馆自动化建设中的应用[J]．聊城大学学报，2003(04)

[5]王冬爱，张涛．xml 与数据库映射技术研究[J]．湘潭师范学院学报，2004(10)

[6]陈爱民，习胜丰．xml 与数据库技术在 web 中的应用[J]．湖南城市学院学报，2004(3)

[7]刘汉兴,田绪红,孙微微．基于Web的XML与数据库映射[J]．现代计算机，2002(12)

[8]田原，唐铸文．xml 和数据库之比较与转换．电脑知识与技术[J].2005(12)

[9]张福军．基于 xml 的分布式数据交换平台的研究与实现[J]．科技创新导报，2008(9)

[10]朱勤,陆建新,陈继红．基于XML的异构数据交换技术及其Java实现[J]，2004(11)

[11]丁月华,杨敏等．基于 XML 的异构数据源集成与交换的实现[J]．计算机应用与软件，2006(10)

[12]魏东平,潘向阳．基于 XML 的异构数据的整合与集成模式探讨[J]．内蒙古科技与经济，2005(5)

[13]关辉．异构数据库间数据交换技术研究与实现[J]．电脑知识与技术，2007(19)

[14]雷刚跃．基于 XML 的异构数据库间数据交互技术研究[J]．科学技术与工程，2006(23)

[15]何慧,陈博. 基于XML和JMS的异构数据交换集成的研究[J]. 计算机技术与发展, 2006(12)

[16]陈天煌,邹青梅. 基于 XML 的异构数据库信息共享技术研究[J]. 武汉理工大学学报, 2005(1)

[17]彭其华. 网络环境下基于 XML 的异构数据交换的研究[J]. 西南民族大学学报, 2003(6)

[18]郑丽丽,魏慧,石秀君. 基于 XML 的异构数据交换的研究[J]. 电脑与信息技术, 2007(4)

[19]Y. Xing, M. G. Madden, J. Duggan, and G. Lyons, "Distributed Regression For Heterogeneous Data Sets"[J], Lecture Notes in Computer Science, 2003, Vol. 2810, pp.544－553.(SCI indexed).

[20]Y. Xing, M. G. Madden, J. Duggan, and G. Lyons, "Context-based Distributed Regression in Virtual Organizations", in Proceedings of the ECML/PKDD-2003 Workshop on Parallel and Distributed Computing for Machine Learning[J], Dubrovnik, Croatia, 2003, pp.80－91.

[21]Xing, M. G. Madden, J. Duggan, and G. Lyons, "A Multi-Agent System for Context-based Distributed Data Mining", Technical report, NUIG-IT-170503[J], Department of Information Technology, National University of Ireland, 2003.

[22]Y. Xing, "Context-based Numeric Prediction for Distributed Data with Contextual Heterogeneity"[D], PhD thesis, Department of Information Technology, National University of Ireland, Galway, 2004.

[23]http://www.xml.net.cn/

[24]Hass L.M,Transforming Heterogeneous Data with Database Middleware[J] Beyond Integration .IEEE internet Computing, 1999,107-123

[25]Ana Mercedes、Molina Vargas、XKoaster：A Tool for Catalog Management on Middleware Databases Systems，Electrical and

Computer Engineering Department University of Paerto Rico, 2002, 61-68.

[26]Manuel Rodriguez·Martinez、Nick Roussopoulos, MOCHA: A Self-Extensible Database Middleware System For Distributed Data Sources, SIGMOD Conference, 2000, 213—224.

[27]Joseph Albert, Data Integration in the RODIN Multidatabase System, Computer Sciences Department University of Wisconsin, 2000, 76-98.

[28]Michael Carey、Jerry Kieman、Jayavel Shanmugasundaram, XPERANTO: A Middleware for. Publishing Object-Relational Data as XML Documents, Proceedings of the VLDB Conference, Egypt, September 2000, 412—123.

[29]MichaelClarke、GordonS. Blair、GeoffCoulson[J], An Efficient Component Model for the Construction ofAdaPtive Middlewllre, 2001, 147—154

[30]王长松，秦琴等，数据库应用课程设计案例精编[J].北京:清华大学出版社,2009.3.

[31]沈兆阳,Java与XML数据库整合应用[J].北京:清华大学出版社,2002.

[32]Elliontte Rusty Harold 著，刘文红 赵伟明灯译，Java 语言与 XML 处理教程[J].北京：电子工业出版社，2003.

[33]张涛.基于XML和Web Service异构数据转换的设计与实现[D].中国海洋大学，2009.

[34]李冠宇，刘军，张俊.分布式异构数据集成系统的研究与实现[J].计算机应用研究.1001-3695(2004)03-0096-03

攻读学位期间发表的论文

雷冲，廉迎战. JSF 框架的研究及其应用[J].网络安全技术与应用.2009 年第 5 期

独创性声明

秉承学校严谨的学风与优良的科学道德，本人声明所呈交的论文是我个人在导师的指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包括其他人已经发表或撰写过的研究成果，不包括本人或其他用途使用过的成果。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明，并表示了谢意。

本学位论文成果是本人在广东工业大学读书期间在导师的指导下取得的，论文成果归广东工业大学所有。

申请学位论文与资料若有不实之处，本人承担一切相关责任，特此声明。

论文作者签字：雷冲

指导教师签字：

2010 年 6 月 5 日

致谢

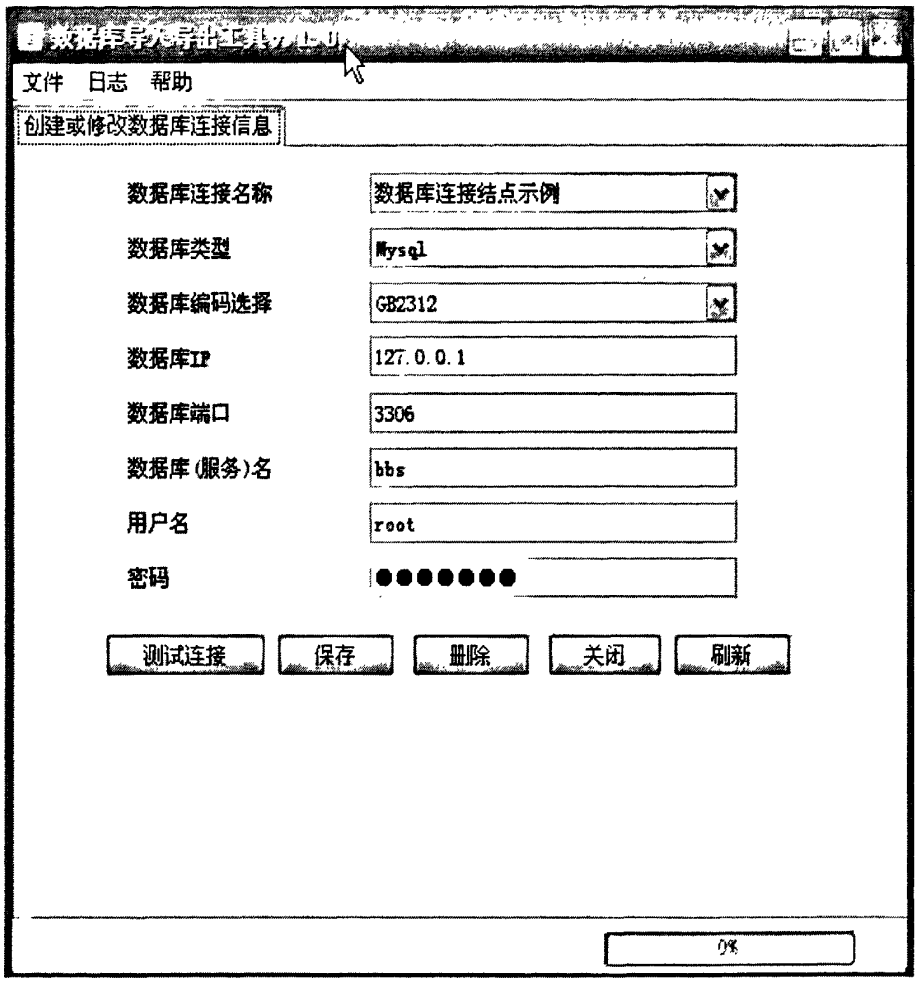
三年的学习生活即将结束，回顾三年的学习生活，感受颇深，收获丰厚。在论文的写作过程中，有很多困难，无论是在理论学习阶段，还是在论文的选题、资料查询、开题、研究和撰写的每一个环节，无不得到导师的悉心指导和帮助。借此机会我向导师表示衷心的感谢！同时，我要感谢广东工业大学授课的各位老师，正是由于他们的传道、授业、解惑，让我学到了专业知识，并从他们身上学到了如何求知治学、如何为人处事。同时我也要感谢我的同学给予我的帮助，他们为我撰写论文提供了不少建议和帮助。我要感谢，非常感谢我的导师廉迎战老师。他为人随和热情，治学严谨细心。他宽容大度的胸怀也深深地影响了我，在论文的写作和措辞等方面他也总会以“专业标准”严格要求你，从选题、定题开始，一直到最后论文的反复修改、润色，廉老师始终认真负责地给予我深刻而细致地指导，帮助我开拓研究思路，精心点拨、热忱鼓励。正是廉老师的无私帮助与热忱鼓励，我的毕业论文才能够得以顺利完成，谢谢廉老师！还要感谢三年的大学生活，感谢我的家人和那些永远也不能忘记的朋友，他们的支持与情感，是我永远的财富。最后，衷心感谢于百忙之中评阅论文的各位老师专家、教授！

谢谢！

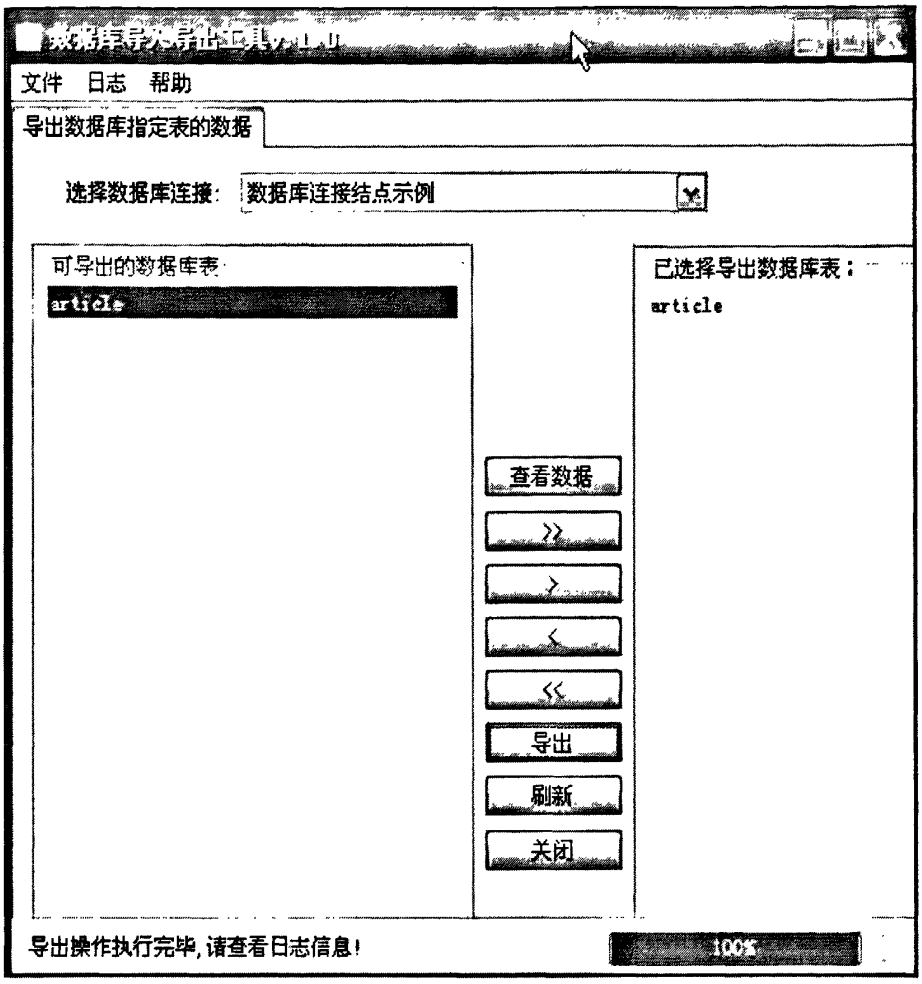
雷冲

2010 年 6 月 4

附录



附录图1 数据库连接测试



附录图 2 导出数据库指定表的数据