

EDI-XML 通用数据转换构件系统的设计与实现^{*}

任洪敏^{1,2} 钱乐秋¹ 蔡国钧²

(上海海事大学计算机系 上海 200135)¹ (复旦大学计算机科学系软件工程实验室 上海 200433)²

摘要 本文分析了基于 EDI 和 XML 技术的电子商务系统应用集成问题,描述了一个 EDI-XML 通用数据转换构件系统的设计原则和软件架构,论述了该架构方案中 EDI-XML 数据模式的定义机制,提出了一个 EDI-XML 通用数据转换的实现算法。

关键词 EDI, XML, 企业应用集成, 构件系统

Design and Implementation of EDI-XML General Data Conversion Component System

REN Hong-Min¹ CAI Guo-Jun² QAIN Le-Qiu¹

(Department of Computer Science and Technology, Shanghai Maritime University, Shanghai 200135)¹

(Department of Computer Science and Technology, Fudan University, Shanghai 200433)²

Abstract This paper analyzes the problem on enterprise application integration of e-commerce systems based on EDI and XML technologies, presents a general EDI-XML data conversion component system's design principles and software architecture, discusses EDI-XML data Schema definition mechanism of the architectural solution, finally describes an implementation algorithm for EDI-XML general data conversion.

Keywords EDI, XML, Enterprise application integration, Component system

1 引言

电子数据交换 EDI(Electronic Data Interchange)是电子商务(EC)尤其是 EC B2B(Business to Business)的支撑技术,广泛应用于制造业、运输业、零售业以及卫生保健和政府部门,以支持在不同的单位之间进行数据的电子化交换和处理。EDI 系统的执行标准包括 EDIFACT 和 ANSI X.12,并通过增值网络或直接连接进行安全、可靠的商务信息传输,现已具有广大的用户基础和长期的应用历史,是国际上大型公司开展电子商务的基础^[1]。

但 EDI 系统安装、运行费用高昂,并建立在固定事务集合的基础之上,缺乏灵活性和扩展性,又 EDI 报文消息可读性差,应用程序难以开发和维护。因此,实施 EDI 系统的多是供应链上的核心企业,许多中小型企业无力承受。结果,实施 EDI 系统的核心企业也不能完全做到节省费用,因为供应链上大量的中小型企业并没有使用 EDI 系统。

XML 是互连网络的新型数据交换标准,能够清晰地描述信息的逻辑结构并提供安全支持,是 Web 数据交换的标准。基于 Web 和 XML 技术的数据交换方案廉价、灵活且易于扩充,得到众多工具、环境与技术的支持,现已成为众多组织建设信息系统和实施电子商务的主流选择方案^[2]。

因此,在信息系统的建设中,经常需要整合组织内已有的基于 EDI 的信息系统和新开发的基于 XML 的信息系统,以既保护遗产系统的投资又提供快速应变的商务系统。同时,更迫切地需要整合组织内基于 EDI 技术的信息系统与 Web 上大量基于 XML 技术的中小型企业的信息系统,以灵活地拓展和变更自己的业务。故整合基于 EDI 与基于 XML 的信

息系统成为企业应用集成 EAI(Enterprise Application Intergration)广泛面临的一个重要课题^[3]。

据此,本文基于软件构件技术^[4],设计和开发了 EDI-XML General Converter 构件系统,提供一个统一的解决方案,支持基于 EDI 和 XML 技术的商务系统双向整合,以积极推进电子商务和信息化建设在中小企业中的开展和实施。

2 EDI-XML General Converter 的系统架构

EDI-XML General Converter 旨在实现一个通用的 EDI 报文与 XML 数据的表示、转换、检测与展示的构件系统,需要遵循软件构件技术的思想和原则,进行良好的架构设计。

2.1 系统架构设计原则和实施策略

EDI-XML General Converter 的架构设计^[5]遵循三个原则:

(1)通用性原则。EDI-XML General Converter 必须能够处理各种 EDI 数据报文,包括不同记录数目、不同字段数目、数据记录间的各种复杂逻辑关系、甚至多个 EDI 报文组合而成的复合报文。据此,EDI-XML General Converter 提供了数据模式定义构件 SchemaDefiner,运用 XML Schema 严格定义报文数据模式的描述机制,提供 SchemaParser 构件负责数据模式的解析,以支持 EDI 数据报文的多样性和复杂性。

(2)可靠性原则。EDI-XML General Converter 支持系统应用集成,EDI、XML 数据转换通常自动流转执行,因此,整个系统必须具备良好的可靠性。据此,EDI-XML General Converter 基于数据模式定义数据的属性,并提供了数据错误处理构件、报文数据检测构件,负责转换过程中报文数据的检测和分析,并对错误的报文数据生成日志,指明错误发生的位

^{*}上海市科委人才基金资助项目(05R214206),上海市教委青年基金支助项目(2004116),上海海事大学重点学科基金(XL0101-1)。任洪敏博士,副教授,主要研究领域为软件复用、软件构件技术、软件架构技术、形式化方法。

置和原因,保证分析后的报文转换时不会发生异常情况,提高了系统的可靠性。

(3)易用性原则。软件构件必须提供简单的接口,且易于使用。EDI-XML General Converter 基于三个策略,提升其易用性:(a) 基于 Web Service 技术封装为服务构件,支持跨平台、跨语言集成和调用。(b) 基于文件和文件系统提供使用接口,基本接口参数是数据模式文件、数据源文件、目标文件和 XML 文件目录。(c) 基于 Web 方式和 GUI,提供数据模式定义和 XML 数据展示功能,数据展示构件自动提取 XML 数据结构,并生成用户可自定义的、灵活的数据查询和定位机

制。

2.2 构件系统的层次架构

EDI-XML General Converter 构件系统包括 7 个软件构件:数据模式定义构件 SchemaDefiner、数据错误处理构件 DataErrorProcessor、模式解析构件 SchemaParser、数据分析构件 DataChecker、负责数据转换与展示的 XML Converter、EDI Converter、XML Presenter 构件。整个系统基于严格的分层策略组织和实现上述构件,架构清晰,接口简单,具有良好的易用性和可扩充性,具体如图 1。

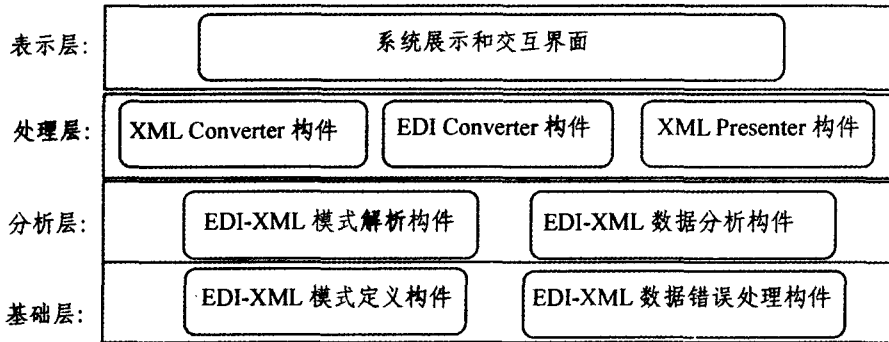


图 1 EDI-XML General Converter 的层次架构

整个系统分为四层。底层是基础层,负责 EDI 模式的定义和数据异常处理。分析层负责执行数据模式解析和报文数据分析。处理层负责执行数据的转换和展示处理,包括 EDI 与 XML 的双向转换和 XML 数据的展示处理。顶层是系统表示层,提供系统界面,负责人机交互。

3 EDI-XML 数据模式定义

数据模式定义是 EDI-XML General Converter 构件系统的基础,EDI-XML 通用数据分析和双向转换皆基于其相应的数据模式定义。EDI-XML General Converter 运用 XML Schema 严格规约数据模式的定义语言。下面给出该模式定

义的基础和其内容主体。

3.1 EDI 报文数据简介

EDIFACT 是 EDI 报文的国际标准。一个 EDIFACT 报文包含若干数据记录,一条记录包括若干字段,记录、字段之间运用分割符进行分割。每条记录具有类型标识符,并且记录间具有复杂的逻辑蕴涵关系,部分记录、字段可以选择或重复。

EDIFACT 数据报文的具体语法参见相关标准。现通过一个航运物流中具体的 EDIFACT 报文实例节选(图 2),展示 EDI 报文的规范,为 EDI-XML 数据模式定义奠定基础。

```
00:BAPLIE:BAYPLAN:9:132208081:13220556-9:0508181124'
10:H3TM:YM ORCHID:PA:37E:09::20050817:20050000:NONE::NONE'
11:YML:YML'
50:AKLU2695111:22G1:E:0491412:.....:2000.0:0.0:KL'
52:GBFXT:FELIXSTOWE:CNSHA:SHANGHAI:CNSHA:SHANGHAI'
50:AKLU2696972:22G1:F:0130302:.....:17200.0:0.0:KL'
52:GBFXT:FELIXSTOWE:CNSHA:SHANGHAI:CNSHA:SHANGHAI'
99:1446'
```

图 2 EDIFACT 报文实例节选

该报文片段包括标识符为“00”、“10”、“11”、“50”、“52”、“99”等六种记录类型,共有 8 条数据记录。其中标识符为“52”的记录逻辑上从属于“50”标识的记录,并且它们可以重复多次。

3.2 EDI-XML 数据模式定义

EDI-XML General Converter 运用 XML 文件定义 EDI 报文的数据模式。EDI-XML 数据模式定义的顶层元素是 <EDISchema>,其属性 Content 标明整个数据报文的内容,其它属性包括该报文中转义字符、空串等的定义。<EDISchema>元素下辖数据报文中数据记录的定义,每条数据记录定义中再包括其组成字段的定义。

数据记录定义规定该数据记录的属性,具体包括:

- (1)Id:数据记录的标识符。
- (2)Class:该记录在 XML 数据中对应元素的标记名称。
- (3)ParentID:父记录 Id。通过该属性指明记录间的层次蕴涵关系,最顶层记录的父记录 Id 为 Null。
- (4)FieldNo:组成该记录字段的数目。
- (5)KeyField:该记录关键字段的编号。
- (6)Seperator:记录中字段间的分割符。
- (7)Order:该记录类型的排位序号。一条父记录拥有多个子记录的时候,子记录的 Order 指明了该子记录在其中的所处的位置序号。该序号用于 XML 数据向 EDI 数据转换时

控制 EDI 报文数据的位置布局,以生成符合规范的 EDI 报文。

(8)MinOccur:该记录类型数据出现的最小次数。如果为 0,则可以不出现。

(9)MaxOccur:该记录类型数据允许出现的最大次数。

EDI-XML 数据模式中字段的定义规定了该字段的属性,具体包括:

(1)No:该字段的编号。

(2)Name:该字段在 XML 数据中的标记名字。字段在生成 XML 时是作为相应记录的属性处理,Name 即为该属性

的名字。

(3)Type:该字段的数据类型。系统的数据类型包括字符串类型、数值类型、日期类型等。

(4)Length:该字段的长度。

(5)SubLength:数值类型小数位的长度,缺省值为 0。

(6)Null:取值是“Y”或“N”,指出该字段是否可以空。

(7)Format:指明该字段的数据展示格式,特别是日期数据,存在多种规定的格式。

(8)Maxoccur:指明字段重复出现的最大次数,用于处理重复字段,缺省值为 1。

```
<EDISchema Content="COARRI">
  <Record Id="00" Class="Head" ParentID="Null" FieldNo="6" KeyField="1" Seperator=":" Order="1"
  MinOccur="1" MaxOccur="1">
    <Field No="1" Name="MessageType" Type="String" Length="6" Null="N"></Field>
    .....<!-- 省略的其它字段定义 -->
    <Field No="6" Name="Time" Type="Date" Length="12" Format="CCYYMMDD" Null="N"></Field>
    .....<!-- 省略的其它记录定义 -->
  </Record>
</EDISchema>
```

图 3 EDI-XML 数据模式定义实例节选

4 EDI-XML 数据转换算法

EDI-XML 数据双向转换算法需要进行数据格式处理、转义字符处理、空串字符处理、超长字段处理等工作。EDI-XML General Converter 对上述处理工作进行了良好的封装,便于用户继承和重载。

XML Converter 构件负责执行 XML 数据报文向 EDI 数

据报文的转换。其转换算法的主要思想是接受 EDI-XML 数据模式信息的制导,读取和查找 XML 数据文件,获取信息,组装成为 EDI 的记录并写入 EDI 报文。XML 数据的层次结构反映了数据记录间的逻辑蕴涵结构,故 XML Converter 构件的转换算法运用递归策略处理数据转换中的层次结构,算法简洁清晰。

```
//变量定义:
int PreRecord; //刚处理记录;
int CurRecord; //现正处理记录;
Stack LevelStack; //追踪逻辑层次的栈

读入第一条EDI数据记录并构造和写入XML文件,置PreRecord为该记录; //建立后续循环处理环境。

while (EDI报文中尚能读取一条数据记录)
{
  读入一条数据记录,置CurRecordId为该记录Id;
  if (EDI-XML模式中没有该记录定义) 则省略该记录;
  else if (CurRecord是PreRecord的子节点)
  {
    构造并写出CurRecord的数据; //其结束元素后续写入
    LevelStack.push(PreRecord);
  }
  else if (CurRecord与PreRecord具有相同的父结点)
  {
    写入PreRecord的结束元素;
    构造并写出CurRecord的Element结点;
  }
  else //否则,前一记录是嵌套最底层的记录,已经结束,找当前记录的父记录
  {
    LevelStack循环弹出,并写出弹出记录的结束元素,直到到CurRecord的父记录;
    写入该兄弟结点的结束元素; //弹出结束时即回到了CurRecord的兄弟,并结束了其嵌套的记录
    构造并写出CurRecord的Element结点;
  }
  PreRecord=CurRecord; //设置当前记录为上一记录,因为其将成为历史
}
```

图 4 EDI 数据报文转换算法

EDI Converter 构件负责执行 EDI 数据报文向 XML 数据报文的转换。EDI 数据报文在格式上是个平板文件,整个

数据文件的逻辑结构由其数据模式定义文件反映,并且在生 (下转第 160 页)



图4 NUST603人脸数据库部分归一化人脸图像

本试验中的训练数据集和测试数据集均随机生成,分别从每类中取 $\vartheta=3,4,5$ 构成训练样本集。每次实验中的训练样本集均随机产生,数据库中训练样本集之外的数据构成测试样本集。表2为参数 $C=1$ 时,在不同训练样本的数目情况下三种方法的识别率比较。从表2可以看出,SUMSDC方法的识别率高于其余的两种方法,这说明该方法所抽取的鉴别信息的有效性高于其余的两种方法。图5是样本数为5、参数 $C=1$ 时,在不同的类别数目情况下,三种方法的识别率比较。

表2 三种方法在 NUST603 人脸数据库上的识别率(%)比较

方法	3	4	5
SUMSDC	95.98	97.74	98.75
MSDC	94.05	95.37	96.67
PCA	90.33	93.40	94.79

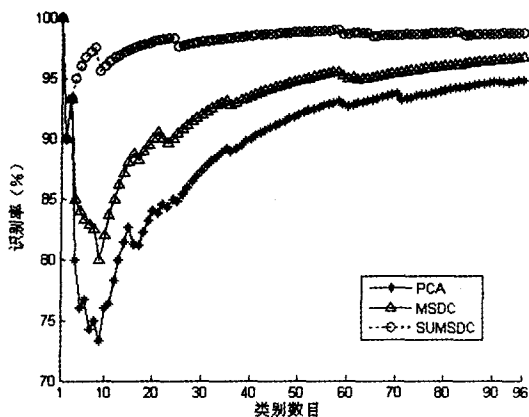


图5 三种方法在 NUST603 人脸库上不同类别数目下的识别率比较图

从表1与表2可以看出,所提出的 SUMSDC 特征抽取方

法的识别率最高。图2与图5则说明了该方法有较好的鲁棒性。从以上的两种人脸库的实验可以看出,SUMSDC方法选择了一组具有统计相关性的鉴别矢量作为投影轴,经过投影变换后,使得特征分量之间具有统计不相关性,从而消除了特征分量之间信息的冗余,提高了识别率。

结束语 本文基于最大散度差鉴别准则,提出了求解具有统计不相关性的最佳鉴别矢量集的计算方法。该方法消除了特征分量之间的统计相关性,提高了人脸的识别率。同时,本文揭示了最大散度差鉴别准则与 Fisher 准则的最佳鉴别矢量集之间的关系。

参考文献

- 1 Fukunaga K. Introduction to statistical pattern recognition [M]. New York, Academic Press, 1990
- 2 边肇祺,张学工. 模式识别[M]. 第2版. 北京:清华大学出版社,1999
- 3 Pentland A. Looking at people; sensing for ubiquitous and wearable computing [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(1): 107~119
- 4 Belhumeur P N, Hespanha J P, Kriegman D J. Eigenfaces vs fisherfaces: recognition using class specific linear projection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997, 19(7):711~720
- 5 Fisher R A. The use of multiple measurements in taxonomic problem [J]. Annals of Eugenics, 1936, 7(1):179~188
- 6 Song F X, Cheng K, Yang J Y, et al. Maximum scatter difference, large margin linear projection and support vector machines. Acta Automatica Sinica, 2004, 30(6): 890~896
- 7 宋枫溪,杨静宇,刘树海,等. 基于多类散度差人脸表示方法. 自动化学报, 2006, 32(3): 378~385
- 8 Jin Z, Yang J Y, Hu Z S, et al. Face recognition based on the uncorrelated discriminant transformation [J]. Pattern Recognition, 2001, 34(7):1405~1416
- 9 陈绵书,陈贺新,刘伟. 一种新的求解无相关鉴别矢量集方法, 2004, 27(7): 913~917
- 10 ORL face database. <http://www.uk.research.att.com/facedatabase.html>

(上接第131页)

成 XML 数据报文的时候,须借助 XML 文件数据元素的嵌套结构反映该逻辑结构。但是 EDI 数据报文可能具有大量的数据,基于性能考虑,不能在内存中依据 DOM 模型生成该 XML 文件并灵活地构造其层次结构。故必须基于文本和顺序读写的方式转换并生成 XML 数据报文。因此,生成算法必须顺序读取 EDI 报文数据记录,并把该记录数据构造为一个 XML 的 Element 结点和其相应属性输出到 XML 文件,但何时输出该元素结点的结束结点 EndElement 以构成 XML 报文正确的嵌套层次结构,却是一个需要仔细考虑的问题,因为 EDI 数据报文通常存在复杂嵌套的逻辑结构。据此,EDI Converter 建立一个栈追踪和控制结束结点的生成,其伪码算法表达如图4。以数据记录元素的读写作为基本操作,该算法的时间复杂度是 $O(n)$,其中 n 是数据记录的数目。

结论 基于 EDI 与基于 XML 的信息系统的集成是 EAI 实施中广泛面临的一个重要课题。本文基于软件构件技术,

设计和实现了 EDI-XML General Converter 构件系统,该系统包含一组彼此独立、但相互配合的软件构件,具有良好的通用性、可靠性和易用性,能有效支持基于 EDI 和 XML 技术的信息系统的整合,现已在上海航运物流领域中获得了良好的运用。进一步的工作包括更多 EDI 标准体系的支持和实时监控转换模式的支持。

参考文献

- 1 龚炳铃. EDI 与电子商务[M]. 北京:清华大学出版社,1999
- 2 肖富军,陈文层,胡运发. 基于 XML 技术的电子数据交换. 计算机工程, 2000, 26(9): 129~131
- 3 王晓耘,李春颖. 基于 XML/RDF 的供应链管理商务信息交换. 计算机工程, 2004, 30(15): 195~197
- 4 杨美清,梅宏,李克勤. 软件复用与软件构件技术. 电子学报, 1999, 27(2): 68~75
- 5 巴斯. 软件构架实践. 清华大学出版社, 2004