# 一种数据自动转化软件的构思与设计*

金 鑫,王晓英,魏绍荣,解 辉
(青海大学 计算机技术与应用系,青海 西宁 810016)

**摘 要**:针对实际工作环境中数据文件格式繁多,结构与标准不统一,与既定数据库之间互转化难的问题,提出了一种通用的、新型的结构化数据与关系数据库互转化的思路,并从理论和结构上对该思路进行了构思与设计,形成了一套新的能够支持多种文件格式的结构化数据文件到指定关系数据库的全自动互转化软件系统,并利用软件复用等思想和技术在当前已有的可靠工程组件基础上,以Java Web开发技术对模块进行了初步实现,取得了较好的预期效果,为今后解决类似异样数据互转化问题提供了思路和途径,在工程实现上也具有一定参考与借鉴价值。

**关键词**:结构化数据;数据转化;关系数据库;软件设计

**中图分类号**:TP317 **文献标识码**:A **文章编号**:1674-7720(2014)01-0014-04

# The design and implement of the software for automated mutual-transforming between structural data files and RDB

Jin Xin, Wang Xiaoying, Wei Shaorong, Xie Hui
(Department of Computer Technology and Application, Qinghai University, Qinghai 810016, China)

**Abstract**: Because there are many kinds files used to record structural data in the real life, and those files have different forms and standards, so many problems and difficulties appear when we want do transforming between those structural data files and a relational-DB. This paper presented a method for automated mutual-transforming between structural data files and a relational-DB, and also the conception and design's introduce of the method. We even have used the technologies about Java-web and software-reuse and developed a software model for our design. The model shows that the method we presented is feasible and can do transforming well between the certain files and RDB. This paper could offer some advises and reference for similar problems.

**Key words**: structural data; data transform; RDB; software design

实际工作中记录数据的数据格式及文件多种多样,这些多样性在给用户提供了较多的选择自由时,也潜在的带来了差异与统一的问题。在现实的生活与工作中,最典型的便是多种结构化数据文件统一化的困难。

大多数与数据库相关的数据迁移及转化研究都关注于数据库系统或信息系统后台控制方面,比如参考文献[1][2][3][4]讨论的是企业数据库后台迁移方面的问题;也有文献讨论了诸如XML之类更结构化数据向数据库转化和迁移的问题,比如参考文献[5][6][7];也有一些关注于通用或专用数据库迁移或转化中间件研究,比如参考文献[8][9][10]。但他们的目标基本都在实现数据库后台管理和减轻DBA的负担,而不是把目标指向一线的数据录入业务人员。

然而在实际工作中,一线人员面临着很大的异种数据录入困难问题。由于地区差异、个人差异、单位差异和信息化软件更迭差异等导致本来应该是同一数据结构和格式的数据经过诸多差异影响后,出现了统一难或难统一的问题,给工作人员带来了新的矛盾和工作难度。比如,在参与某地区的审计系统信息化培训时发现,这些数据操作员平时用的软件可能是微软的Excel2003、Excel2007、Excel2010,也可能是Access2003、Access2007、Access2010,也有些选择Sqlserver系列,也有的选择Orical系列,甚至有些农村、偏远地区和少数民族地区计算机操作能力较低的人员使用word或TXT格式的纯文本文件来记录数据。而他们处理的本是同一格式审计数据,却出现了不同格式

* 基金项目:国家自然科学基金项目(No.60963005)

和版本的数据文件，导致最终的汇总难度很大。

另外，在诸如学校、机关单位、企业或商业系统的网站或管理信息系统中，也面临类似问题。系统要求用户按照规定的格式输入大量的业务数据。而这些数据此前可能因个人喜好问题以不同的格式和文件类型存在于用户本地机器上。当要输入时，用户首先会期盼系统能够批量处理；其次期望系统能够处理的文件格式是当前的数据文件格式。这种依赖于个人需求的期望和系统单一或有限支持功能形成了明显矛盾，而矛盾的根源就是数据文件的多样性和不统一。

在百度搜索中以“数据库数据转换”，“数据文件转化为数据库”之类的关键字进行搜索，可以找到数量巨大的与终端用户数据输入工作有关的提问、帖子和文章，这反映了在现实工作中一线用户所面临的诸多问题，恰如前面所分析。

其实正是选择的多样性导致了记录数据的文件类型多样、格式不统一等现象，进而使得许多在业务上需要综合和汇总的数据文件因格式和文件类型差异而无法合并或难于合并的尴尬现象出现。

本文所述内容正是为了解决这类问题而进行的一些研究和开发工作及成果。

## 1 问题的解决思路

解决实际工作中所产生的不同格式、结构和文件类型的统一化问题显得很急迫和重要。如何解决这个问题，大体上可分为两类思路。

（1）对用户的政策化管理方式

即利用行政文件或命令的形式，要求所有用户采用某种统一格式或规范对数据进行处理与存储，保障数据格式和文件类型的统一。

优点：对信息系统的技术依赖性较低，对原系统或指定系统需要少了甚至不需要改动，系统改造成本低。

缺点：行政性过强，导致用户需要改变或学习指定的数据文件处理技术，有一定的行政命令执行时间段，同时还需要一定时间的统一化培训和学习时间，从行政命令下发到系统可用的时间间隔较大；行政管理成本大、人工成本高、用户体验度较低，系统可扩展性较弱。

（2）对系统进行技术升级或改进

升级（或设计和开发）可支持多种数据文件处理功能的新系统，使系统可以自动对现有的异种数据文件进行正确处理与存储的能力，从而保障数据格式和文件类型的一致性。

优点：系统一旦升级或开发完成，在不是很长的部署周期后，系统即可投入使用，部署距离实用时间较短；系统自动进行异种类型数据文件的统一化处理，各终端用户不需考虑学习新软件或数据文件处理技术，用户体验度较高；对行政依赖相对较小，管理成本相对较低；系统可扩展性较强。

缺点：对信息技术依赖程度相对较高，需要投入一定的开发成本，并组织工程技术队伍对原有系统进行升级，或重新设计开发新系统；需要一定的开发周期。

## 2 软件模块结构的设计

根据业务实际需求，采取了第（2）种思路的方式。首先查阅了与主题有关的许多文献资料，有一些从软件结构上做了一些研究工作，如参考文献[2][8][10]，但重点不是面向用户数据服务；也已经有一些如参考文献[11][12][13]用来实现一些指定文件向数据库的转化，然而这些文献仅仅从编程角度对问题进行了说明，在软件的架构设计和软件工程的层面上，却鲜有较系统和规范的设计与实现见于报道。

鉴于此，根据数据转化的需求，本文设计和实现了一种能够自动转化软件的结构。

### 2.1 转化软件模块的总体结构

根据实际业务需求设计出整个数据处理概念模型如图1所示。在该概念模型中，可以看到，对于可支持的多种类型的数据文件，比如Excel系列、TXT文件、Access系列、MySQL等类型，经期望的“数据统一化综合处理模块”处理后，可以转化为某指定后台数据库中的标准形式，从而达到了自动标准化与统一化的目地。整个模型中“数据统一化综合处理模块”是核心，也是系统要解决的重点问题，依据此要设计出期望的软件模块。

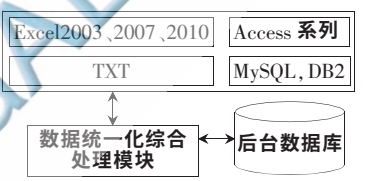


图1 转化软件模块的总体结构

### 2.2 核心模块结构图设计

有了第一步的顶层模型设计后，工作重点将聚集在核心模块“数据统一化综合处理模块”，整个数据转化模块中的核心模块，组成结构与设计问题上。对该模块的组成结构设计如图2所示。

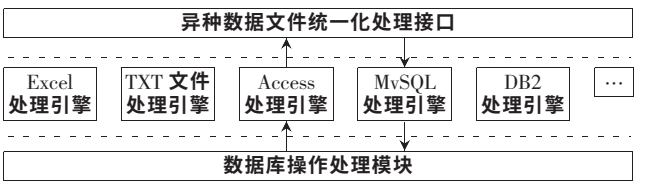


图2 核心模块结构设计图

图2所述模块通过统一化处理接口接收或发送用户数据，通过图中数据操作处理模块与数据库进行互操作。比如用户经过web界面将数据文件提交于处理模块，模块中异种数据文件统一化处理接口根据文件类型将文件选择性的传递给对应的文件处理引擎。经专门的引擎处理后，形成相对统一化的数据，然后将这些数据再写入到统一的数据库中去，从而最终形成统一标准的结构化数据信息。

以用户提交一个 TXT 文本最终系统转化成 MySQL 数据库数据为例。这个过程如下：

用户通过 UI（比如 web 页面形式）提交文件→“异种数据文件统一接口”进行预处理和选择对应引擎→对应“引擎”进行转化分析与处理→“数据库操作处理模块”实现与数据库的互操作。用户下载的过程与此相反，不再赘述。

## 3 软件模块的设计与实现

本文采用基于 Java 的 Web 系统开发技术，平台采用 Java2SE、eclipse、dreamweaver8.0 作为开发平台，以 resin3.1.9 为 Web 服务器运行平台，后台数据库采用。因此，在模块的实现中，所要解决的就是其他类型的数据文件与 MySQL 数据库的互转化问题。

### 3.1 TXT 与 MySQL 的互转化

利用从后台数据库读取数据后，经处理然后生成 TXT 文件，再将该文件传送到客户端。为了解决 TXT 与 MySQL 的相互转化，对 TXT 文本的格式做了要求，第一行表示表头，其他每一行代表一条记录，每个字段使用制表符（Tab）隔开，如图 3 所示。

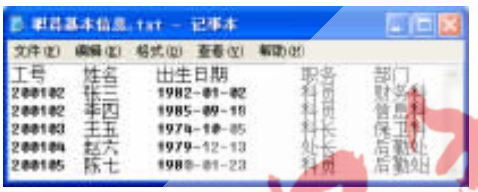


图 3 规范化的 TXT 文本内容

这样就可以利用 Java 操纵本地文件的功能，方便地根据制表符 Tab 和回车换行符（\r\n）来读取 TXT 文件中的每一行记录和每一个记录字段。再顺利读取了表头后，就可以形成用于执行数据库插入的 PrepareStatement 对象，然后再逐行读入每行的各字段形成 Preparestatement 对象的执行参数，然后通过 exceute 函数执行，从而实现写入数据库的目的。

从数据库向 TXT 文本转化时，先利用 JDBC 从数据库中读出记录，在利用 java 的本地文件读写功能，以格式化的形式向文件中写入每一个记录。以上图所示为例，具体写入语句格式为：

“工号”字段值+“\t”+“姓名”字段值+“\t”+“出生日期”字段值+“\t”+“职务”字段值+“\t”+“部门”字段值+“\r\n”

对应数据视图读写完毕后，将形成一个载有所有记录数据的 TXT 文档，将此文档再利用远程下载方式传送到客户端，从而实现 MySQL 数据库向 TXT 文本的转化功能。

### 3.2 Excel 2003 与 MySQL 的互转化

利用类似 TXT 文件处理的思路，加入对 Excel 进行针对性操作的功能，可以将后台数据库中所生成视图中数据按照顺序和结构依次写入到所生成 Excel 文件的对应单元格中，主要以 Excel2003 为基础进行试验。为了提高开发效率和降低成本，采取了在 Jakarta 的 POI 基础上来进行二次开发的策略。具体过程是在工程中引入“poi.jar”包，在此基础上开发出读写 Excel 文件中数据的应用模块，并结合 JDBC，实现后台数据库与 Excel 文件之间的相互自动转化技术。

整个业务处理流程如图 4 所示，其原理与 3.1 中 TXT 文档的处理相似，除处理的文件类型不同外，主要是流程中间引入了用于精细化操作 Excel 的 POI 模块。

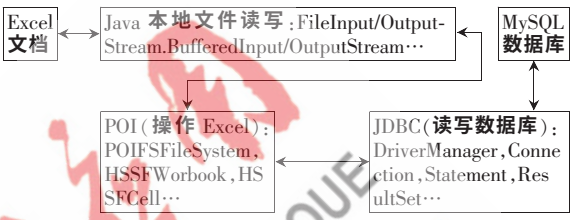


图 4 Excel 与 MySQL 互转化流程图

在这个部分的实现过程中，对于 POI 的熟悉掌握及在其之上的二次开发工作是关键。尤其涉及到单元格精确操作的对象与函数的使用很重要。

### 3.3 从其他数据库向 MySQL 的转化

目前为止，主要实现的是前面两种非数据库文件向关系数据库的转化。对于诸如 Access，DB2，SqlServer 等系列数据库文件向 MySQL 的转换，也做了相应的开发工作。以 Access 与 MySQL 的转化为例，主要实现思想如图 5 所示。

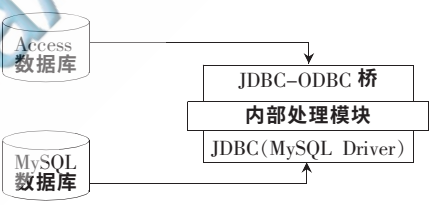


图 5 Access-MySQL 转化原理图

为实现 Aceess 与 MySQL 的相互转化，创建了一个类 DB_Tran_MySQL_Access，在该类中用 Java JDBC 操纵 MySQL，利用 JDBC-ODBC 桥操纵 Access，然后根据业务需求利用 Java 代码实现二者转化，部分核心代码如下：

```
//连接源数据库
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection connAccess=
                    DriverManager.getConnection(
"jdbc:odbc:target","","");Statement stmt=connAccess.
createStatement();ResultSet rs=stmt.executeQuery("select*
from employee");
//连接目标数据库
Class.forName("com.mysql.jdbc.Driver");
String url="jdbc:mysql:          //127.0.0.1:3306/test"
+"?characterSetResults=GBK";
Connection connMySQL=
```

```
DriverManager.getConnection(
    url,"tester","pass123");
//执行记录的转移
PreparedStatement pstmt=
connMySQL.prepareStatement(
"insert into employee(id,name,department,salary)
values(?,?,?,?)");
while(rs.next()){//循环装入数据
    pstmt.setInt(1,rs.getInt("id"));pstmt.setString(2,
rs.getString("name"));
pstmt.setString(3,rs.getString("department"));
pstmt.setDouble(4,rs.getDouble("salary"));pstmt.
executeUpdate();}……
```

上述是 Aceess 向 MySQL 转化的主要实现思路和过程。其他数据库文件与 MySQL 的自动互转化实现与此过程基本相似，不再赘述。

## 4 系统的性能评估

### 4.1 测试用数据说明

为了检验系统对用户的业务提升效率，将开发后的原型系统部署在某省的审计单位进行测试。该单位在全省各市、州和县均有下属机构或单位，并且由于多民族和地区发展不均衡，业务人员的信息化操作水平参差不齐，差异较大，如文章开头部分所述，是比较典型的面临数据文件多样与不一致问题的单位。再进行协调后，选取了其某一季度的业务数据用本文所述系统进行处理的效果对比，所处理数据文件数量及类型情况如表 1 所示（涉及到的记录大概 30 000 多条）：

表 1 测试数据文件数量及比例

| 数据文件类型 | 文件数量 | 所占比例/% |
|---|---|---|
| excel | 284 | 64 |
| txt | 21 | 3.4 |
| access | 94 | 21.1 |
| sqlsever | 51 | 11.5 |

### 4.2 两种数据录入方式的准确率

由于该单位之前的录入工作大部分为手工操作，为了更好说明问题，分别利用人工和原型系统对 2 000 条标准的结构化数据记录进行了转化准确率对比，表中只是对 txt 和 excel 两种典型文件记录进行了测试，而对于 access 和 sqlsever 没有测试，主要是因为这两种格式人工输入效率类似 excel，自动输入方式用户也可能选择一些数据库转化工具，效率与系统大致相同。对比数据如表 2 所示：

表 2 两种数据录入方式的准确率

| 记录条数准确率 / 文件类型 | excel | | txt | |
|---|---|---|---|---|
| | 手工/% | 自动/% | 手工/% | 自动/% |
| 100 | 95 | 100 | 90 | 100 |
| 200 | 90 | 100 | 80 | 100 |
| 500 | 83 | 100 | 75 | 100 |
| 2 000 | 68 | 100 | 65 | 100 |

由于人工错误率较高，那么在实际的使用中，可能会进行多次的检查，意味着需要更多的人工和时间。

### 4.3 部署前后的效果对比

对所选取的数据文件利用文中所述系统进行了自动转换，并与其以往的方式所消耗时间进行了对比。为了更真实的分析本系统对业务的积极影响，这里所谓处理时间不是单指计算机中的处理时间，而是包含了整个文件准备、输入、处理和快速检查等步骤，因为这个时间是真正的业务处理过程所消耗。

所测数据文件中，几个类型的数据文件所消耗时间均不到一个小时，这里为了对比方便，都视为处理时间为 1 个小时。而之前该单位利用其以往的处理方式进行处理所消耗时间就相对大的多。对其业务量和劳动时间进行了统计，部署前后的对比数据如表 3 所示：

表 3 部署前后的效果对比

| 数据文件类型 | 以往处理时间（小时 * 人） | 自动处理时间（小时 * 人） | 效率比 |
|---|---|---|---|
| excel | 5×8 | 1 | 40:1 |
| txt | 3×8 | 1 | 24:1 |
| access | 1×8 | 1 | 8:1 |
| sqlsever | 1×8 | 1 | 8:1 |

上述的对比中忽略了人工输入错误率高的因素。否则人工处理的时间成本会更高。

为了更直观地看到前后效果对比，对上述表格中的数据以直方图表示如图 6 所示。

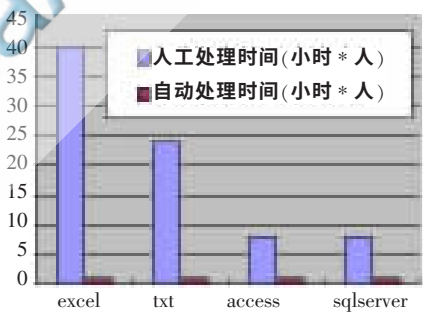


图 6 部署前后效率直方图

从图可见，所设计和实现的异种数据文件自动转化系统确实大大提高了数据的统一和规范化的处理效率，很实际的解决了用户单位所面临的具体业务困难，节省了业务成本。

总体上，本文所提出的软件设计思路较新颖的从系统层面提出了一种通用异种数据转化为统一数据库数据的解决方法；并从中间件的角度构思和设计了一个可实现异种数据转化的软件模块的全新架构；对系统进行初步工程实现，验证了设计的可行性；原型系统表现出较好的自动化处理效果，为解决用户存在的实际困难和问题提供了一个有效途径。同时本文所提出的软件思想和设计也不仅仅局限于文中所述问题，也可以对诸如

Web 系统中的不同网页格式的统一化处理、对于网络中的不同配置文件的统一化处理、以及其他类似问题的处理都有很实际的参考和借鉴价值。

当然，由于系统初步实现，还存在一些问题，如：(1) txt 文本和 excel 文件内容必须是全结构化的，因此对文件内容格式要求较严格，导致用户对文件的操作受到一定限制；(2)系统对文件内容的格式依赖程度较高，如果文件中不慎出现格式问题，将可能导致数据转化的错误与失败；(3)系统稳定性等方面的问题。不足之处，将在今后的工作中继续关注和解决，期望能有更完善的结构和软件实现。

参考文献

[1] 熊华平. 大型异构数据库数据迁移系统的研究与应用[J]. 计算机应用与软件，2012(10)：178-181.

[2] 唐彬.异构数据库数据迁移中间件设计[D].兰州：兰州理工大学，2011.

[3] 张小波. 基于协同数据库的数据迁移模型研究与实现[J]. 计算机工程与设计，2005，26(5)：1220-1222，1301.

[4] Xiong Junjun. Key Technology Research of Migration from Informix to Oracle[J]. Computer Engineering, 2012, 7(14): 52-55.

[5] 黄根平，郭绍忠，陈海勇，等.数据集成中 XML 模式和关系模式映射模型研究[J].信息工程大学学报，2009(4)：527-531.

[6] 郑仕勇. 基于 XML 和中间件的异构数据库数据迁移的研究与应用[D].桂林：桂林理工大学，2010.

[7] Li Aimin, Tan Xianhai. Research on conversion of unstructured data to structured data based on XML technology [J]. Railway computer application, 2012(10): 12-16.

[8] 刘如九. 一种通用的多数据库间数据抽取方法及应用[J]. 北京交通大学学报(自然科学版)，2008(4)：14-18.

[9] 韦琳，袁泉，霍剑青，等.E-learning 非结构化数据管理系统的构建与实现[J].中国科学技术大学学报，2010，6(40)：307-311.

[10] 徐燕.信息系统中的通用数据迁移工具的研究与设计[J]. 计算机与现代化，2010(6)：156-158.

[11] http://www.sj00.com/article/578/579/2006/200605238423.html [EB/OL][2006-05-23].

[12] http://www.docin.com/p-244646744.html[EB/OL][2011-08-16].

[13] http://wenda.tianya.cn/question/4d2c1166d69e0f8f [EB/OL] [2009-05-10].

(收稿日期：2013-09-12)

作者简介：

金鑫，男，1965 年生，本科，讲师，主要研究方向：计算机应用，数据库。