

基于 SOA 的银行非现场稽核系统中 ETL 设计与实现

王 珊

(深圳职业技术学院 电子与信息工程学院, 广东 深圳 518055)

摘 要: 本文基于 SOA 设计思想, 采用 XML 技术实现异构数据源的通用接口访问方式, 提出一种采用配置文件来增强系统灵活性的方法, 该方法提高了系统的可扩展性, 稳定地完成 ETL 功能. DCT 系统也可方便扩展用于其他行业数据仓库的 ETL 任务.

关键词: SOA; ETL; DCT; 非现场稽核

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2010)08-0261-04

Design and Realization of Bank Non - scene Auditing ETL Based on SOA

WANG Shan

(School of Electronic and Information Engineering, Shenzhen Vocational Technical College, Shenzhen 518055, China)

Abstract: A self-developed ETL tool which is named DCT system is designed and realized under the background of the commercial bank non-scene auditing. The ETL system based on SOA is proposed in this paper, the XML technology is added for implementing the general access interface to heterogeneous multi-data sources. It is also enhanced the flexibility by using the configuration file. DCT system is convinced for the further development and expansion. Practical applications indicate that DCT system has a high flexibility, scalability. ETL functionality is stable completed. The implementation of DCT for bank data warehouse system is put forward.

Key words: SOA; extract-transform-load (ETL); data copy tool (DCT); bank non - scene auditing

1 引言

随着银行业的高速发展和激烈竞争, 商业银行急需利用数据仓库技术, 对各业务系统所积累的数据进行提炼和分析, 实现对各项业务合规性的自动化审计及对银行经营风险的监控和预警.

ETL 工具^[1-2]的未来发展方向之一是打包应用集成. 文中提出一个基于 SOA 的 ETL 的设计思想并开发了一个 ETL 工具——DCT (Data Copy Tool), 以期探索将 SOA 的模块化、低耦合高内聚、接口等思想应用在 ETL 设计当中, 使 ETL 朝 SOA 方向发展, 即尽量不动代码, 而是用类似“软件配置”的思想实现需求变化^[3-4].

2 基于 SOA 的非现场稽核系统体系结构设计

银行要求建设的系统要具有高效稳定, 可扩充性、先进性等性能, 以实现电子化银行监管体系和建立稽核方法知识库, 提高稽核工作效率和质量, 增强风险防范能力. 根据上述目标, 结合 SOA 思想设计了如图 1 的逻辑架构.

系统划分为后台子系统和前台子系统两大部分.

后台子系统完成数据采集和数据整理的工作, 实现数据采集的自动化; 前台子系统则实现审计工作的各项功能. 通过前后台子系统的分离, 在后台子

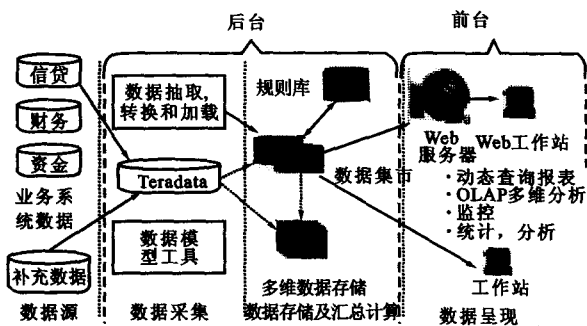


图1 非现场稽核系统的逻辑架构

系统中提供灵活的多种系统接口,实现了系统的底层无关性,即系统无需作程序修改,就可以连接不同平台上的业务系统及其数据结构,适应商业银行现有的业务系统运行模式。

前台子系统应用分为 Web 服务和数据库应用服务两部分,Web 服务直接面对用户,所有实现的程序和服务都集中在一个服务器上,用户通过浏览器来使用本系统。数据库应用服务对客户的数据进行采集、处理和展现。

3 DCT 设计思想与功能模块

3.1 基于 SOA 的 ETL 架构设计

文中提出开发的 ETL 应向 SOA 方向靠,就是尽量不动代码,而用类似“软件配置”的思想实现需求变化,这样做的优点有:

(1) 开发向标准化发展,开发团队风格容易统一,也方便未来维护;(2)开发成本降低,用“配置”比“修改代码”的代价永远要低;(3)维护成本大大降低,维护人员甚至可以不用看代码,就能维护。

故此,按照商业银行非现场稽核信息系统项目数据采集的要求,基于 SOA 的思想,提出了如图 2 的模块化 ETL 设计架构。

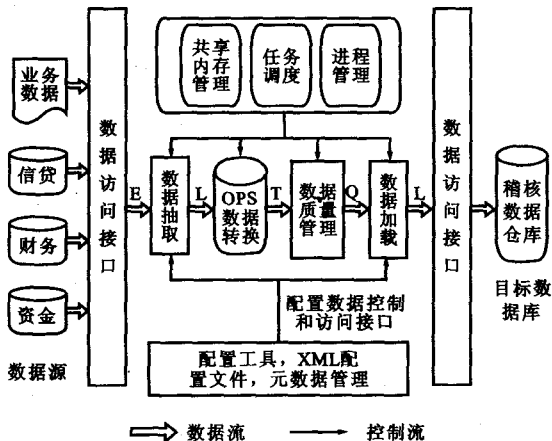


图2 ETL 架构

3.1.1 数据访问接口

银行已建立的业务系统选择了不同的数据库产品,数据来源千差万别,有数据库文件,非数据库数据源,如文本文件等。为使系统具有更好的扩展性,系统采用了 XML 技术,用 XML 语言描述数据源,实现异构数据源的通用访问接口。ETL 程序只需从 XML 配置文件中根据不同的标记读取各种数据源的信息,建立数据连接。

3.1.2 参数化配置文件

采用参数化的方式,通过可视化的 MapForce 软件配置各类源及目标数据库、数据表、抽取周期、源数据到目标数据的转换规则,数据清洗等内容,实现元数据的动态管理。当配置文件被重新设定或更改后,ETL 程序的循环读取会将内容实时更新到 ETL 服务器的缓冲区中,ETL 数据抽取程序根据缓冲区中内容开始新的任务,这样的运行方式使系统具有了很好的自适应能力。

3.1.3 ODS 层设计

由于稽核系统要从很多业务系统中提取数据,各业务系统的数据结构不一,而且编码方式也大相径庭。因此有必要在数据进入稽核 DW 之前进行统一编码等,文中增加了 ODS(数据存储区)层,将采集的数据放入 ODS,然后在 ODS 中对数据进行清洗、转换,经过一系列的处理之后,ODS 中的数据质量会显著提高,基本符合稽核系统的需要。在设计 ODS 时,将 ODS 中大部分表的结构设计成和稽核系统对应表同构,部分表因特殊原因无法设计成与稽核系统同构的,也通过增加冗余字段等方式使二者基本同构。这样更利于稽核 DW 的生成。

3.2 DCT 的功能模块设计

DCT 开发结构分为共享内存及信号量管理、任务调度、进程管理、数据抽取、数据装入、数据转换、数据质量管理模块。

作为核心,共享内存及信号管理、任务调度、进程管理这几个模块是固定的,共享内存模块通过创建静态参数的设置,完成共享内存及信号量的管理。任务调度模块的功能是从共享内存区中读取需要执行的任务,根据数据装入关系表生成所有的步骤序列,逐一执行这些步骤,直至步骤结束。这三个模块通过共享内存区及信号量进行数据交换、同步。

数据抽取、数据装入及数据转换,数据质量管理提供函数接口,其他模块通过接口调用它们,这几个模块是可以扩展的,如图 3 所示。

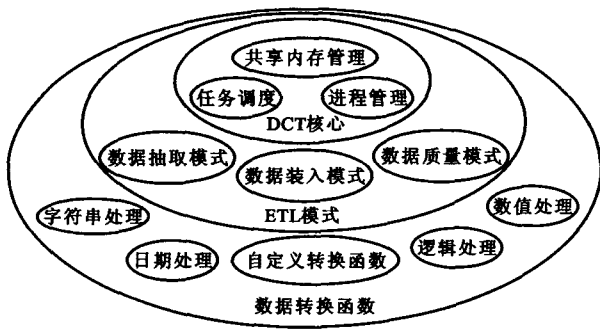


图3 DCT 开发结构

4 DCT 实现的关键技术

DCT 是一个 ETL 框架,采用 XML 技术实现异构数据源的通用接口访问方式,通过配置文件中的参数执行 ETL 任务,针对不同的应用系统,允许开发出不同的数据抽取、数据装入、数据转换函数并加入到 DCT 框架中.这样系统就具有很好的适应性和扩展性.下面阐述实现 DCT 良好扩展性的主要技术.

4.1 基于 XML 的配置文件技术

XML(ExTensible Markup Language)是一种简单的数据存储语言,能以一种开放的自我描述方式定义数据结构,在描述数据内容的同时能突出对结构的描述,XML 可使应用系统在无须了解特定数据表示格式的情况下,轻松地交换数据^[5].

DCT 的配置文件都采用 XML 语言,创建主配置文件来管理共享内存区及信号量;创建静态路由表配置文件来定义网络拓扑结构;创建数据装入关系表配置文件来定义数据的 ETL 方式(即数据从哪个源通过什么途径到哪个目标);创建起始及终止任务配置文件来定义所有任务开始之前及所有任务结束之后需要批量执行的命令.以下是 DCT 主配置文件代码一个片段.

```
<DCT>
  <IPCS comment="IPCS 参数">
    <!-- 共享内存区 KEY,信号量 KEY -->
    <ShmFile> $ PRJDIR/etc/DCT.SHM</ShmFile>
    <SemFile> $ PRJDIR/etc/DCT.SEM</SemFile>
  </IPCS>
  .....
  <!-- 数据装入关系表 -->
  <ELRRelation> $ PRJDIR/etc/etl.xml</ELRRelation>
  <!-- 能同时运行的最大进程数 -->
  <MaxProc>10</MaxProc>
</DCT>
```

DCT 配置文件利用 XML 文档做为数据交换载体,以文件方式实现数据交换.

4.2 ETL 模式开发机制

通过开发 ETL 模式,可以使得 DCT 支持更多的应用.开发 ETL 模式概括为以下几个步骤:

(1) 在相应目录中开发新的 ETL 模式的处理代码.

(2) 在 DCT 系统中注册新的 ETL 模式,包括申明函数原型、注册到函数数组中等.

(3) 在 ETL 配置文件中使使用新的 ETL 模式.

下面以数据抽取模式为例详细介绍 ETL 模式的开发过程.

4.2.1 开发新的 ETL 模式代码

每一种数据抽取格式都对应于一个抽取模块.每一个抽取模块设计了三个函数接口:

```
int extractOpen(DCTCTRL *c, ETLITEM *e);
int extractFetch(DCTCTRL *c, ETLITEM *e, unsigned char **b, size_t *l, ETLITEM s);
int extractClose(DCTCTRL *c, ETLITEM *e);
```

其中 extractOpen() 函数用于连接数据库或者打开文件等数据抽取源.

extractFetch() 函数用于逐条读入数据源,并将其写入到缓冲区中,其中的 b 是一个指向数据缓冲区的指针,数据缓冲区在本函数 extractFetch() 中分配内存,但使用后调用者必须释放该内存.

extractClose() 函数用于关闭游标,数据源、释放资源.

另外因为同一个模块在一个 ETL 任务^[6-7]中可能被多个步骤使用,所以不能通过静态变量的方式来存储全局信息.因此必须使用扩展参数结构体变量,并将其存储在变量 e->ExtRun 中.

每一个模块都可以自定义与扩展参数相关的结构体.为了让所有的模块程序具有相似的程序结构,我们设计了在定义扩展参数结构体后,使用以下宏定义来统一扩展参数名称:

```
#define EXTPARSTR struct _extpar _e_sep-
text _dbtb
```

4.2.2 注册新的 ETL 模式

在 DCT 系统中注册新的 ETL 模式,包括申明函数原型、注册到函数数组中.

(1) 在头文件 \$PRJDIR/src/lib.src/etlfunc.h 中申明 ETL 模式的函数原型,例如:

```
/* 将带分隔符的文本串装入到数据库中 */
#define ETL_L_SEPTTEXT_DBTB "L|DATABASE|
```

SEP-TEXT"

```
int etlSepText _ DBTB _ Open (DCTCTRL *,
ETLITEM *);
int etlSepText _ DBTB _ Put (DCTCTRL *,
ETLITEM *,
unsigned char **, size_t *, ETLITEM);
int etlSepText _ DBTB _ Close (DCTCTRL *,
ETLITEM *);
```

(2)在相应源代码文件中注册 ETL 的三个函数到 DCT 的函数数组中,例如用 ETL _ REGOPENFUN 宏注册 Open()函数,如下所示:

```
ETL _ REGOPENFUN(ETL _ E _ SEPTXT _ FILE,
etlSepText _ File _ Open)
```

4.2.3 配置文件使用新的 ETL 模式

使用新的 ETL 模式,只需要在 etl.xml 中指定 ELFunction 元素及对应的 Extension 元素即可,主要代码如下:

```
<ELFunction>L/DATABASE/SEP-TEXT/TRANS-
FER</ELFunction>
<Extension>
  <mfd remark="MapForce 文档名称">
    $ PRJDIR/etc/mfd/dcttest.mfd
  </mfd>
  <xsdpath remark="数据结构定义文档搜索路径">
    $ PRJDIR/etc/xsd/arnis
  </xsdpath>
</Extension>
```

4.3 转换函数开发机制

DCT 提供数据转换函数数组,并预定义了一些系统的转换函数,对于用户或者特殊的数据转换,可由用户在实际中自己定义那些转换函数.对于自定义转换函数,只需按 DCT 设计的接口标准编写数据转换函数,并将函数在函数数组中注册,就可在 DCT 的 ETL 任务中使用该函数.

4.3.1 数据转换函数接口标准设计

在 DCT 中为所有的数据转换函数设计的函数原型如下:

```
int TRF _ foo(int argc, char ** argv);
```

其中:

argc: 参数个数.

argv: 参数数组.argv[0]始终表示函数的出参.

返回值:

0: 转换函数执行成功.

<>0: 转换函数执行错误,错误信息写入日志

文件.

4.3.2 函数原型的申明

在 DCT 头文件 \$ PRJDIR/src/lib. src/data-tran.h 中申明函数原型,例如:

```
DAT _ DCLTRANFUN(GetBussDate) /* 取得当前营业日期 */
```

4.3.3 注册到 DCT 数据转换函数数组中

在相应源程序的 dctInitDataTranFun 函数中注册数据转换函数,如:

```
DAT _ REGTRANFUN("GetBussDate", Get-
BussDate)
```

4.3.4 在 MapForce 中使用自定义数据转换函数

用 MapForce 定义数据转换规则,再由 DCT 来解析这些规则,并在数据转换模块中根据转换规则调用相应的数据转换函数.

首先将函数库导入到 MapForce 中,然后,就可以在数据转换定义中使用该函数了.

5 运行效果

系统经过三年多的运行,运行稳定,不轻易发生系统错误,不轻易发生程序死机.每天数据抽取、转换、装载的时间不超过 3h,每月数据抽取、转换、装载的时间不超过 2 天,对一个 6000 万条记录的数据库表进行数据遍历查询,响应时间为 1 分钟.实现了监管部门对银行业务活动的全面、持续、高效、准确地监控.

6 结束语

文中针对某商业银行非现场稽核系统中 ETL 部分,基于 SOA 设计思想提出了可复用的 ETL 架构设计策略并实现一个 ETL 框架工具 DCT. DCT 系统采用模块化设计,使用 XML 技术实现了通用接口,解决了数据来源复杂和数据导入效率等问题,使用参数化配置文件来增强系统的灵活性,提供二次开发的标准接口,便于针对新的应用开发新的 ETL 模式,也便于针对新的转换规则开发新的数据转换模块,这样的设计保证了 DCT 系统具有更好的灵活性、系统扩展能力和自适应能力.

参考文献:

- [1] 王磊,李一凡,赵怀慈. 银联数据仓库系统中 ETL 的设计和实现[J]. 微电子学与计算机, 2007, 24(5): 66 - 72.

(下转第 268 页)

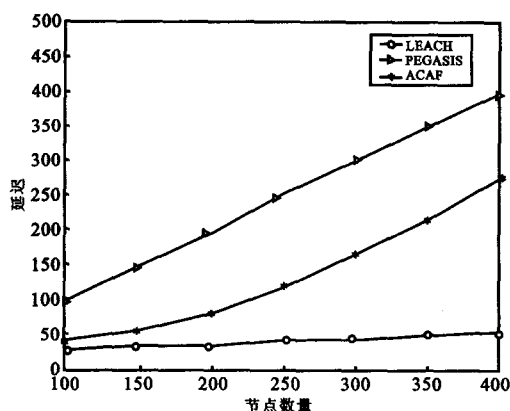


图3 延迟的比较

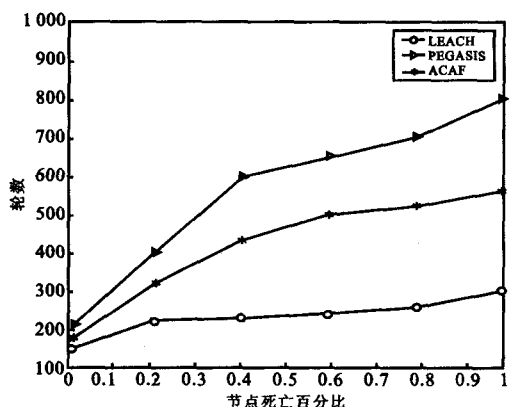


图4 节点生命周期的比较

传输实时性要求和方便簇首节点部署需求的自适应分簇算法,其核心思想是将网络根据距离中心 SINK 节点的远近划分为多层,在网络形成初期,各层簇首节点与 SINK 直接通信;而当采集的数据达到稳定可以进行火险的阶段预测后,各层自适应采用多跳通信,传感器节点感知到的环境数据由所属层簇首转发给内层簇首,节省传感器能量延长网络生命周期。文中采用簇首为超级节点以及网络初期各簇首单跳通信的方法,从理论上分析可以有效减少成簇的消息开销,以及早期提高簇与 SINK 节

点的通信效率,但并未涉及底层的具体通信协议,下一步工作将逐步进行完善,并从通信效率上进行更细致的比较分析。

参考文献:

- [1] 唐晓燕,孟宪宇,易浩若. 林火蔓延模型及蔓延模拟的研究进展[J]. 北京林业大学学报, 2002(1):87.
- [2] 李光辉,赵军,王智. 基于无线传感器网络的森林火灾监测预警系统[J]. 传感技术学报, 2006(6):2760.
- [3] 任丰原,黄海宁,林闯. 无线传感器网络[J]. 软件学报, 2003(7):1282-1291.
- [4] 王淑娟,吕振,于跃. 基于信息融合技术的火灾监控系统[J]. 电气应用, 2008(3):45-47.
- [5] Li J, Mohapatra P. Analytical modeling and mitigation techniques for the energy hole problems in sensor networks[J]. Pervasive and Mobile Computing, 2007,3(3): 233-254.
- [6] Olariu S, Stojmenovic I. Design guidelines for maximizing lifetime and avoiding energy holes in sensor networks with uniform distribution and uniform reporting[C]// Proceedings of IEEE INFOCOM. Spain, Barcelona, 2006:1-12.
- [7] Soro S, Heinzelman W. Prolonging the lifetime of wireless sensor networks via unequal clustering[C]// Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium. [S. l.]: IEEE Press, 2005: 236-240.
- [8] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocols for wireless microsensor networks[C]// Proceedings of International Conference on Systems Science. Hawaii, 2000.

作者简介:

雷建军 男,(1969-),副教授.研究方向为计算机网络与并行计算.

万润泽 男,(1981-),硕士,讲师.研究方向为无线传感器网络.

(上接第 264 页)

- [2] Thomas Jorg, Stefan De loch. Towards generating ETL processes for incremental loading[C]// Proceedings of the 2008 international symposium on Database engineering & applications. New York, USA: ACM, 2008:101-110.
- [3] 叶宇风. 基于 SOA 的企业应用集成研究[J]. 微电子学与计算机, 2006,23(5):211-213.
- [4] 魏东,陈晓江,房鼎益. 基于 SOA 体系结构的软件开发方法研究[J]. 微电子学与计算机, 2005,22(6):73-76.
- [5] 顾天竺,沈洁,陈晓红. 基于 XML 的异构数据集成模式

的研究[J]. 计算机应用研究, 2007,24(4):94-96.

- [6] 王珊,陈琨. ETL 中基于贪婪算法的任务调度方法研究[J]. 微电子学与计算机, 2009,26(7):130-133.
- [7] Hao Fan, Hao Gui. Study on heterogeneous data integration issues in Wb environments [C] // WiCom 2007. [S. l.]: IEEE, 2007:3755-3758.

作者简介:

王珊 女,(1970-),硕士,讲师.研究方向为数据仓库.