

ETL 应用优化设计与实现研究

张 靖^{1,2}, 雷 航¹, 唐雪飞¹, 孟帅朋¹

(1 电子科技大学 计算机科学与工程学院, 四川 成都 610054; 2 攀枝花学院 网络中心, 四川 攀枝花 617000)

摘 要: 分析和研究了传统 ETL 工作流程处理过程, 采用 JAVA 技术, 提出了 ETL 工作流程优化方案, 对传统 ETL 性能问题提出了优化办法, 并进行优化处理, 很好地解决了数据集成处理过程数据融合、数据一致化和数据同步更新等重要问题, 减轻了瓶颈制约, 节约了处理时间开销。实际应用证明, 优化后的 ETL 流程方案有效地减少了传统 ETL 工作流的时间花费, 执行效率高, 具有良好的实践应用价值。

关键词: ETL; 优化; 设计; 效率; 性能

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2012)04-0134-04

Study on the Optimization Design and Implementation of ETL Application

ZHANG Jing^{1,2}, LEI Hang¹, TANG Xue-fei¹, MENG Shuai-peng¹

(1 School of Computer Science and Engineering, Univ. of Electron. Sci. & Tech. of China, Chengdu 610054, China;
2 Campus Network Center, Panzhihua University, Panzhihua 617000, China)

Abstract: The processing procedure of the traditional ETL work flow is analyzed and studied. Based on JAVA, the optimized scheme of ETL work flow is presented, and it optimizes the traditional ETL performance, perfectly solves the problems about data fusion, data consistency and data synchronous update in the data integration processing procedure, and reduces bottleneck restriction and the processing time. The application proves that the optimized ETL flow scheme effectively reduces the time of the traditional ETL workflow with favorable application value.

Key words: Extract-Transform-Load; optimization; design; efficiency; performance

1 引言

ETL(Extract-Transform-Load)是数据源端不同数据库或异构数据源的数据经过抽取、转换和加载到目的端的过程,在数据处理领域已成为一种成熟的技术^[1]。但是面对不同应用系统大量的数据处理实践过程^[2],传统的 ETL 还存在着一些问题:(1) 流程方案在数据集成处理方案过程中无法很好地实现异构数据源系统数据的融合、数据一致化处理和数据同步更新;(2) 处理性能还不适应各种系统异构的海量数据处理,及时性和一致性差,提取数据存在一定的瓶颈。因此,要根据数据集成平台的实际需要,优化的 ETL 流程方案,改进 ETL 处理过程以及

方法,通过 ETL 配置文件增强系统灵活性,利用系统处理并行操作,充分发挥系统的性能,提高系统资源利用率,减少处理时间,提高 ETL 性能和效率。

2 ETL workflows 方案设计

传统 ETL 模型流程方案架构设计和操作方式简单且易实现,但由于是顺序执行,对于源数据库很多且来自区域各异、数据量很大时,大大降低了资源利用率和软件工作效率,大量数据将进入等待状态,加重了数据存储中心负担,也增加了程序员对 ETL 流程的可控性。优化 ETL 模型工作流程如图 1 所示,对数据转换过程进行了分解,分别是数据验证、清洗和过滤、整合、预计算和数据聚集,加入了临时

使用的数据库(内存数据库)使得数据频繁操作可快速的在内存中完成,减少反复数据库 I/O 操作带来的时间花费,整体提高 ETL 处理数据速度^[3]。

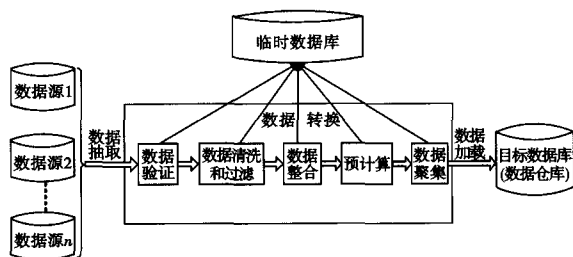


图1 优化后的 ETL 工作流程

3 ETL 优化设计和实现

ETL 优化设计主要是使用优化的 SQL 语句、增量抽取流程优化、数据清洗流程设计、数据同步算法设计、元数据管理框架设计、ETL 工作流多线程设计,使用内存数据库技术,进行数据一致性处理。

3.1 增量抽取流程

增量抽取^[4]使用时间戳的方式,基于快照比较的变化数据捕获方式。根据这种增量抽取方法的要求,在源数据表上增加一个时间戳字段。当系统要更新数据表时修改时间戳字段值,增量数据抽取时,提取出表中的时间戳字段值,对比系统时间和这个时间戳字段值,决定所要抽取的那部分数据。具体流程设计如图2所示。

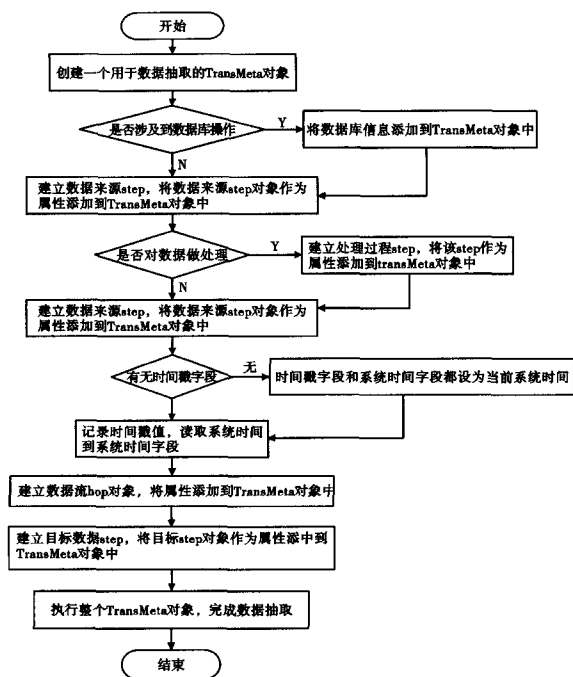


图2 数据抽取流程

3.2 数据清洗

数据清洗要处理的数据主要是不完整的数据、错误的的数据、重复的数据,实现流程设计如3所示。

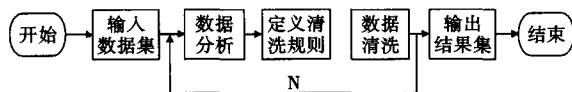


图3 数据清洗实现流程

3.3 数据同步

在源数据库上添加一个时间戳字段,在进行数据转换过后的数据表上保留这个时间戳,每进行数据抽取的时候,先读取这个目标数据仓库表的时间戳的最大值,把该值作为参数传给源数据库的相应表,对比这两个时间值来决定要抽取的数据,数据抽取之后同样要保留这个时间戳,并更新源数据库上的时间戳。数据同步过程算法如下。

输入: Smi; StepMeta 接口的一个实例

Sdi; StepData 接口的一个实例

输出: 空

- (1) 从数据集中得到一行, $r = \text{getRow}()$ 。
- (2) 如果 r 为空,则设置步骤完成标志。
- (3) 复制输入内容的元数据到输出元数据里。
- (4) 查询目标数据集到 r_{target} 。
- (5) 判断 r 中关键字对应数据和 r_{target} 中对应数据是否一致,设置 operation 是 'I', 'D' 或是 'U'。
- (6) 判断 operation 值,执行不同的操作, 'I' 对应 insertValues, 'D' 对应 deleteValues, 'U' 对应 updateValues, 三个操作对应三个子处理过程。
- (7) 把 r 内容写到输出域中, $\text{putRow}(\text{rowMeta}, r)$ 。
- (8) 设置步骤完成标志。

3.4 ETL 元数据管理

许多数据应用领域采用元数据^[3],主要是两项工作^[5]: (1) 建立元数据模型框架。设计的元数据模型基于 CWM (Common Warehouse Metamodel, 通用仓库元模型), CWM 构建元数据所需的语法和语义,还包含了 ETL 流程过程所需要的各种元数据模型,通用性、重用性和扩展性好。 (2) 建立数据管理系统。根据 CWM 元模型^[6]的要求,进行了 ETL 元数据管理系统框架的设计,其整体框架图如图4所示。

3.5 ETL 多线程执行模式的设计

让每个节点 Node 运行在一个单独的线程上,两个 Node 直接连接 Edge 提供 n 个缓冲区 buffer。当 Node 向 Edge 写数据时,如信号量 $S > 0$,说明此时 Edge 有空缓冲区,则往里写数据,否则进行等待处理。读取数据的过程与此类似,这个操作过程是经

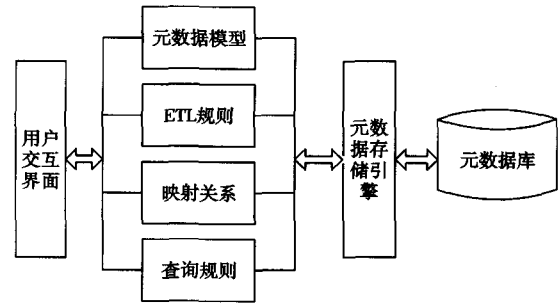


图4 元数据管理系统框架

典的生产者-消费者问题. 设定了三个信号量, 其中两个 empty 和 full 用于解决生产者 Node 和消费者 Node 线程之间的同步问题, mutex 用于两个线程之间互斥的设置, full 初始值为缓冲区数 0, empty 初始值为缓冲区数 N, mutex 初始化为 1, 并发执行流程设计如图 5 所示.

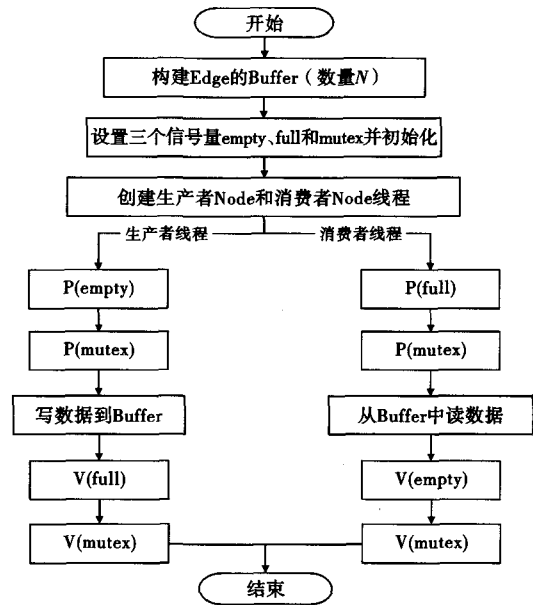


图5 ETL 线程间并发执行处理图

4 ETL 数据仓库设计和实现

4.1 数据仓库设计

ETL 数据仓库整体体系结构如图 6 所示.

4.2 ETL 过程设计与实现

(1) 定义元数据

根据实际业务需求和 ETL 任务的执行过程, ETL 元数据如图 7 所示.

(2) ETL 组织结构和映射关系

对数据的一系列操作活动的组织结构如图 8 所示. 从源数据通过转换规则到目标的过程定义了从

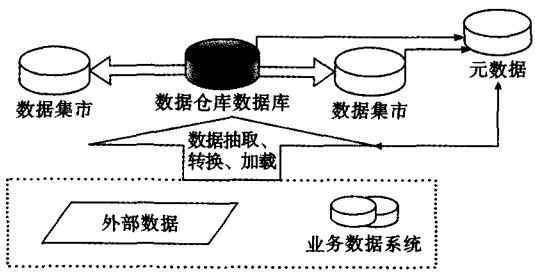


图6 ETL 数据仓库整体架构

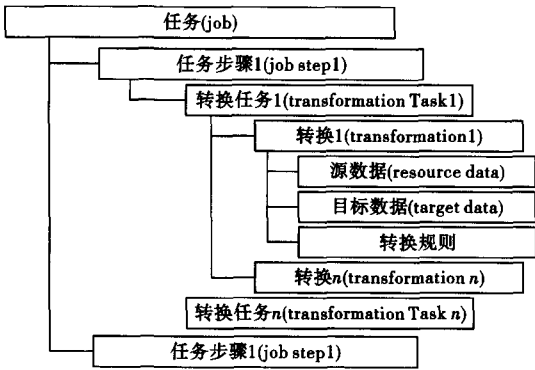


图7 ETL 元数据

源数据到目标的一个映射关系或者映射方法^[8]. 整个关系建立过程的映射关系算法如下:

①在内存数据库中建立步骤对应的表; ②获取用户连接信息 hop(from, to), 即源和目标; ③根据②的信息填写到映射表中; ④根据映射表转换规则函数, 读取转换信息; ⑤映射过程结束.

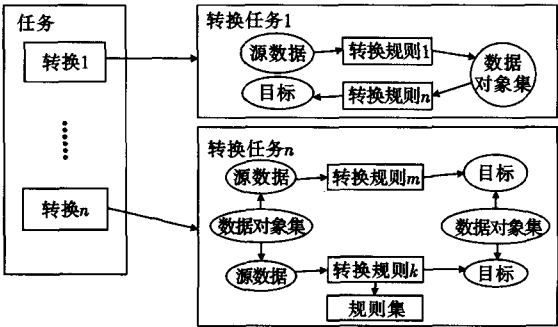


图8 ETL 过程组织结构

(3) 数据类型的映射

各自定义的数据类型不一样, 建立一个整体的数据映射表, 统一由 ETL 来处理.

(4) ETL 流程整体执行算法

①首先获取转换步骤的属性信息, 如字段名、类型、转换格式等. ②根据①信息中的关键字, 获取相关源数据表的属性信息、字段信息和对应的数据源

的信息. ③根据②得到的信息,在内存数据库中创建临时表,把数据源对应的数据添加在内存数据库中. ④根据连接信息,查询下一步节点属性信息,据该信息对刚完成节点输出数据操作,把结果放入连接信息 Buffer 中,提供下一步执行的数据信息. ⑤如果下一步操作没有结束,也就说不是目标表,那么重复④的操作;如果执行结束,为目标表,则将数据加载到目标表中. ⑥完成整个 ETL 过程操作.

5 ETL 效率性能分析

5.1 测试方法

按照需求在两个数据库中设计出指定的测试表,在 MyEclipse 中添加支持数据库操作的 jar 包,并随机生成一批数据并插入表中作为测试用数据.

5.2 测试数据

用 java 程序随机生成一批学生的信息字段(studentid,age,name,sex)和成绩字段(sid,studentid,mark)写入数据库构建测试数据源,数据量达到十万,有一定的代表性和普遍性.上行数据流程操作分别用 Access、Mysql 和 SqlServer2003 为源数据库,目标数据库为 Oracle,下行是 Oracle 为源数据库,Access、Mysql 和 SqlServer2003 分别为目标数据库.

5.3 测试结果

(1)不同数据库之间数据迁移时间

如表 1 所示,在同样的情况下,优化后 ETL 流程整个执行时间小于传统 ETL 流程执行时间,效率远超过了传统的方案.

表 1 不同数据库之间数据迁移执行时间对比

	传统 ETL/s	优化 ETL/s
MySQL 数据库向 Oracle 数据库	559.6	139.9
Access 数据库向 Oracle 数据库	737.5	151.9
SQL.Server 数据库 Oracle 数据库	739.6	132.5
Oracle 数据库 MySql 数据库	273.8	78.9
Oracle 数据库 SQLServer 数据库	499.9	30.6
Oracle 数据库向 Access 数据库	90.5	29.4

(2)整体性能分析

①流程执行时间性能比较

优化后的 ETL 流程有着更高的执行效率.经过粗略加和求平均方法计算,本次过程中传统 ETL 流程执行时间是 569 s,优化后的 ETL 流程执行时间是 88.4 s,比前者节约了 5 倍多的时间花费.

②内存空间使用情况比对

本次 ETL 优化过程中许多操作在内存进行,牺牲了一部分内存空间,同样运行环境下优化后 ETL 流程内存可用数要少于传统 ETL 流程内存可用数,而这部分内存消耗正是新加入的内存数据库.优化后的 ETL 流程多消耗部分内存空间情况下提高了执行效率.

6 结束语

采用 JAVA 技术设计并构建了优化的 ETL 流程方案,整体架构灵活易扩展、支持跨平台,可以很好地解决数据集成处理方案过程中异构数据源系统数据的融合、数据一致化处理和数据同步更新等重要数据问题.对传统 ETL 性能问题提出了优化的方案,并进行优化处理,减少了传统 ETL 不断提取数据带来的瓶颈问题,节约了时间开销.实验验证了在数据处理时优化后的 ETL 流程方案有效地减少了传统 ETL 工作流的时间花费,有优越的执行效率,具有良好的应用价值.

参考文献:

[1] 徐俊刚,裴莹.数据 ETL 研究综述[J].计算机科学,2011,38(4):15-20.

[2] 谭支鹏,冯丹,吴永英,等.基于工作流的数据抽取转换回载[J].华中科技大学报:自然版,2006,34(2):61-63.

[3] 彭璐.基于数据仓库的 ETL 过程优化[J].计算机与数字工程,2010,38(5):166-169.

[4] 戴浩,杨波.ETL 中的数据增量抽取机制研究[J].计算机工程与设计,2009,30(23):5552-5555.

[5] 郑丹青.基于元数据的空间数据仓库 ETL 系统设计与研究[J].吉林师范大学学报:自然科学版,2010(2):43-45.

[6] Gomes P, Farinha J, Trigueiros M J. A data quality metamodel extension to CWM[C]//Proceedings of the 4th Asia-Pacific Conference on Conceptual Modeling, Ballarat: Australian Computer Society, 2007:17-26.

[7] 宋旭东,阎晓岚.数据仓库 ETL 元模型设计[J].计算机仿真,2010,27(9):106-108.

[8] 姚全珠,白敏,黄蔚.基于模型驱动的 ETL 模型映射方法[J].计算机工程,2009,35(19):91-93,96.

作者简介:

张靖男,(1972—),博士研究生,副教授.研究方向为计算机软件和计算机网络.