一种多级数据仓库体系结构和双通道视图更新算法

Multi-level Data Warehouse Architecture and Update Algorithm of Double Channels View

熊忠阳　黄海龙　张玉芳　欧　灵

（重庆大学计算机学院　重庆400044）

Abstract This paper puts forward a new multi-level data warehouse architecture based on WHIPS of Stanford University. First adds sub-data warehouse that has the similar structure of data warehouse. Second brings OLAP history base into multi-level data warehouse for enhancing OLAP query efficiency. Finally for ensuring the data consistency and improving query efficiency, it presents 2 channel view updating algorithm to maintain multiple views on line and parallel realize OLAP query. The improved system separates data updating and OLAP service, avoids the data inconsistency. The establish sub-data warehouse satisfies the department OLAP, increases the working efficiency, and enhances retractility and scalability.

Keywords Data warehouse, Double channel, On-Line Analyze Process, View

1　引言

联机分析处理(On Line Analyze Processing:OLAP)是决策支持系统的重要组成部分，它为企业管理人员提供了利用集成化的数据仓库进行更准确、更完整的信息查询的能力。数据仓库是支持企业或组织决策分析处理的，拥有面向主题的、集成的、相对稳定的数据集合。在数据仓库的联机维护中，只有保证数据仓库数据与数据源的一致性，才能为高层决策人员提供全面一致、有效的分析型战略数据信息。

数据仓库中一般存在多种集成粒度上的视图，视图的元组数据存储在数据仓库中，这种实例化视图一般没有自动更新功能。当数据源发生变化后，数据仓库视图也应该随之发生相应的更新，在数据更新时可能出现以下几个问题：

1)几个视图可能由同一数据源派生，此数据源的更新将带来多视图的同时更新；

2)数据源的广泛分布性，对于分布式的数据仓库存在并发更新的可能；

3)数据仓库多视图的更新和前台的OLAP查询数据之间的一致性问题。

本文着重研究了多视图的OLAP查询与数据仓库数据更新之间的数据一致性问题，对相关数据仓库结构进行了改进并设计出相应的双通道算法，使数据仓库的数据更新和OLAP服务之间的事务分离，避免数据出现不一致，同时使数据仓库的扩展性得到了提高。

2　多级数据仓库体系结构

数据仓库中的下钻(Drill-down)操作在多数情况下需要访问数据源。由于视图建立后的更新维护相对滞后于源数据，当下钻查询用到原始数据和视图数据时，则会出现查询不一致的问题[2]。为了对视图进行联机维护，及时应答用户的OLAP请求，数据源也要参与相应运算，这又会降低数据源端的联机事物处理OLTP性能。为此，在Stanford大学WHIPS系统[1]的基础上，为了解决视图更新时的数据一致性问题以及提高OLAP的效率，本文提出了一种改进的数据仓库结构。

改进之一：在数据源与数据仓库之间加入一个具有数据仓库结构的子仓库，形成多级数据仓库模式，如图1所示。

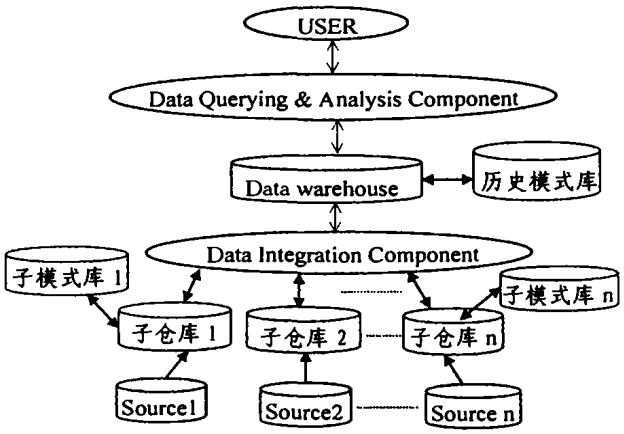

图1　多级数据仓库体系结构

子仓库位于数据源和数据仓库之间，拥有和数据仓库类似的数据结构及关系定义。子仓库向上面的数据仓库发送更新消息及更新数据，向下接收数据源的更新请求。子仓库的引入不仅简化了数据仓库的更新操作，而且简化了数据仓库的视图管理，提高了效率。引入子仓库的主要优点是：

1)保证了数据仓库和OLAP服务数据的一致性，提高了

数据库在实际中的应用。在建立弱点数据库的过程中也遇到了一些现实的问题，这需要我们在今后的工作中加以解决。

参 考 文 献

1　Taimur A, Ivan K, Eugene S. A Taxonomy of Security Faults. In: Proc. of the national computer security conf. 1996

2　Du W, Mathur A P. Categorization of software Errors that led to Security Breaches: [Techinical Report]. CS Department, Purdue University, 1998

3　Kumar S, Spafforf E. A Taxonomy of Common Computer Security Vulnerability Based on their Method of Detection: [Tech Report]. Purdue University, 1995

4　余建斌. 黑客的攻击手段及用户对策. 人民邮电出版社，1998

视图的更新速度，降低了数据仓库更新操作的复杂性。因为子仓库的数据结构与数据仓库类似，同时上传更新数据时已经明确了需要更新的视图对象，分散和缓解了数据仓库视图管理器的工作负荷，提高了系统性能。

2)建立在部门级的子仓库提供部门级的OLAP服务，减少了数据仓库OLAP服务的负荷，而且各级数据仓库容易扩充，OLAP模式库可以根据需要加载。

3)有利于企业以自底向上模式建立数据仓库，可适用于各部门的应用需要，具有较强的灵活性和扩充性。

数据仓库与子仓库之间数据的双向流通性可以使更新数据先经过子仓库处理后再向上传，大大提高了数据仓库的更新效率和OLAP的服务效率，保证了数据的一致性。由于二者在数据结构上的类似性，以及子仓库的更新数据明确了更新的视图对象，因此大大减少了数据仓库更新通道视图管理器的运算量，同时数据仓库向下可以将OLAP查询分解成多个子查询发送给子仓库，子仓库完成子查询后返回结果，极大地提高了效率。

改进之二：为进一步提高OLAP服务的效率，在子仓库之上加入OLAP历史库。历史库中保存OLAP历史查询的条件及结果，其主要作用是为OLAP提供参考，先依据OLAP查询的条件在历史库中搜索条件匹配或近似的查询结果，再进行后续相关的OLAP查询，从而减少OLAP查询的工作量。历史库与数据仓库的结合使多级数据仓库的OLAP服务的效率更高，同时子仓库的历史库存储了部门级OLAP的历史记录，从而也实现了OLAP服务的分级。数据仓库与子仓库之间数据的双向流动性，可以将复杂的、涉及部门级的OLAP请求分散至各部门的子仓库端实现，之后再进行汇总，在一定程度上实现了复杂事务的并行处理。

改进之三：为满足应用层的需要，结合数据仓库模型和网络平台，在网络与数据仓库之间利用JDBC/ODBC进行连接，实现企业多级数据仓库的B/S模式。

经过改进后的多级数据仓库体系结构具有以下特点：

1)多级数据仓库体系结构简化了数据仓库视图更新过程，使数据仓库更集中于满足OLAP服务；

2)多级OLAP的查询服务功能可以满足各部门需求；

3)增加的历史库提高了OLAP查询效率；

4)将数据仓库建立在网络平台上，可以更好地利用和开发网络资源，为经营网络化提供良好的数据平台。

3 双通道更新算法

数据仓库中多视图联机维护已有多种算法，例如ECA[2]、2VNL[3]、STNL[4]、画笔算法[5]等，它们或多或少存在以下问题：

1)数据更新与OLAP服务不易兼顾。决策支持系统主要目标在于OLAP服务，但是有的算法在视图更新上耗费大量运算时间，导致OLAP服务相对滞后；

2)OLAP服务是基于状态版本进行的，有些算法针对更新数据采用多状态和时间戳，导致系统需要同时维护多个版本数据，并且版本之间切换也非常复杂；

3)有些算法对数据仓库并行的任务全部串行化，这一串行化操作在处理某些问题时可能存在隐患。

4)为避免在查询和更新时出现死锁，有些算法将更新操作转化为对临时表的操作，这使得操作复杂化，效率也难尽人意。

基于改进后的多级数据仓库体系结构，以及对OLAP服务的要求，我们认为多视图联机维护应该满足以下三点：1)尽量降低更新事务的消耗，使数据仓库主要集中在如何满足OLAP服务；2)更新效率高且OLAP查询的数据一致，不应出现多个更新时间的数据混杂；3)尽可能减少事务重启，着重提高OLAP查询效率。基于上述思想，我们设计了双通道多视图更新算法。

3.1 算法思想

基于改进的数据仓库结构和多视图维护特性，本文提出了双通道(2 Channel)更新算法，其基本工作原理如图2所示。数据仓库实体化视图上有两个通道：一个通道主要面向OLAP服务，即在OLAP服务态(ON-SERVE)，另一通道则主要面向数据源的数据更新，即在数据更新态(ON-UPDATE)。算法的主要思想如下：

数据仓库实体化视图上有两个通道：OLAP服务通道和数据更新通道。所有OLAP请求在缓冲池中依序排列，按照一定策略进行调度，FCFS(先来先服务)或各种优先级思想都是可以采取的调度策略。数据仓库的OLAP服务通道从缓冲池队列中选取OLAP请求进行处理，首先访问数据仓库的历史模式库，如果有相似的OLAP请求，找到至本次OLAP查询要求时间为止的结果；若没有相似的历史模式，则在数据仓库中进行查询。OLAP结果先保存于历史模式库中，再返回给用户。数据更新通道则主要接受数据源的更新数据，周期性地执行更新操作，其中的数据对用户OLAP服务是透明的。

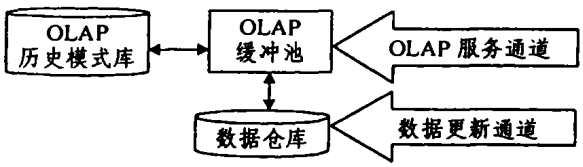

图2 双通道算法示意图

OLAP服务通道在每完成一次OLAP查询后，检查是否有数据更新，执行完数据更新后再进行下一次OLAP查询。为避免主键冲突或死锁，数据更新通道通过附加版本号执行更新数据操作。例如，OLAP服务通道针对0号数据版本执行OLAP查询，在此期间，更新通道提交了3次更新数据，这三次提交的更新数据版本号分别为1、2、3；此时数据仓库中存在版本号分别为0，1，2，3四个版本的数据，而OLAP查询只针对0号版本数据查询，版本号的引入避免了数据更新时出现的数据不一致性问题。当完成一个OLAP查询后，OLAP服务通道检查更新通道，从小版本号开始依次执行数据更新操作。附加的版本号使得实体化视图的更新维护与OLAP查询隔离开来，互不干扰，避免了冲突。

3.2 算法描述

多级数据仓库体系结构和双通道视图维护算法划分成3个部分：

1)OLAP服务通道维护算法：主要实现OLAP调度、OLAP查询接口调用、更新非0版本的数据以及在历史模式库中的匹配；

2)数据更新通道维护算法：主要实现接收后的数据发送视图管理，定时提交更新数据以及附加版本号；

3)通信算法：主要完成I/O操作，实现子仓库和数据仓库的双向数据流动。

本算法具体主要涉及4个算法(A表示OLAP服务通道，

B 表示数据更新通道)：

1)B_Receive(U_data)：参数 U_data 为更新数据，当存储在更新数据缓冲区中的更新数据达到一定数量后向数据仓库或子仓库发送要求更新的数据；

2)A_Update(V_data)：当 B 接收数据后向 A 发送更新通知，参数 V_data 表示更新数据的版本号和时间戳。A 执行 OLAP 服务时 B 可能发送了多次数据更新通知，A 接收更新时完成 B 中所有版本号的更新数据，更新数据前为确定是否有效更新请求，需检查更新记录表 update_record；

3)A_OLAP(OLAP_i)：A 接受用户发送的 OLAP 请求，搜索历史模式库和数据仓库。完成一个 OLAP 查询后允许 A 执行数据更新操作，此时暂停 OLAP 服务；

4)Manage_timer()：实现定时触发检测与通道 A 有关的 OLAP 和数据更新任务。

各算法描述如下：

```
Algorithm B_Receive(U_data)
Begin
    If 有数据更新 then
      在 B 通道中获取新的数据更新版本号 Vi;
      从更新数据缓冲区读取数据到 U_data;
      通过集成器(integrator)返回对应于实体化视图更新的数据
          U_data_new;
      将 U_data_new 写入 B 通道，并且置新的版本号 Vi;
      V_dada ← Vi + timestamp;
      A_update(V_data);//给 A 通道发送数据更新消息
    End if
End

Algorithm A_update(V_data)
Begin
    //查询更新记录表 update_record，是否为有效更新请求
    lb_check = check_in_update_record(Vi , timestamp);
    if lb_check = true then// 已经更新过
      return
    else //尚未更新
      if status_A is true then
        //A 通道状态为 true，允许更新数据；false 时允许 OLAP 查
          询
        While channel_B 的更新队列中有数据
              B_update_A();//从 B 通道中小的版本号开始逐版
                              本更新 A 通道；
                        //删除 B 通道已更新的数据版本和数据；
        End while
        Set_A(false);//设置通道 A 为 false，允许 OLAP 服务；
      End if
    End if
End

Algorithm A_OLAP(OLAP_i)
Begin
    If status_A is false then//如果 A 状态为 false，进行 OLAP 服务
      Get_a_OLAP();//从 OLAP 请求队列中调度一个 OLAP 任
                      务
      分析 OLAP 任务确定搜索方法；
      搜索历史模式库，查找类似的历史 OLAP，并返回结果 result;
      if result is exist then
        返回在历史模式库中的 result;
          Else
              执行 OLAP 查询；
              将 OLAP 查询结果保存于历史模式库；
              返回 OLAP 查询结果；
          End if
          删除 OLAP 队列中的此 OLAP 请求；
          Set_A(true);//设置 channel A 为 true，允许数据更新服
                      务
      End if
End

Algorithm Manage_timer()
Begin
    检查 A 消息队列，获取消息 message;
    if message 是数据更新 then
      获取要更新的数据的 V_data ;
      If V_data 不为空 then//如果 B 通道有更新数据
          A_update(V_data);
      End if
    Else      //OLAP 请求
          If OLAP 请求队列不空 then
              //调度出此 OLAP 请求 OLAP_i;
              Get_a_OLAP(OLAP_i);
                  //从 OLAP 请求队列中调度此 OLAP 任务
              A_OLAP(OLAP_i);
          End if
    End if
End
```

3.3 算法性能评价

基于多级数据仓库体系结构的双通道算法具有以下特点：

1)数据仓库 OLAP 事务与视图更新操作同时进行。视图更新运算被分解在子仓库级完成，由于子仓库的数据量较小，子仓库的视图管理操作不会影响子仓库的 OLAP 查询性能；

2)数据仓库支持联机下查，子仓库与数据仓库的数据双向流动性，在一定程度上减轻了数据仓库的负荷；历史模式数据的提供减少了查询量，提高了查询效率；

3)双通道的版本切换平滑，用户 OLAP 查询始终针对同一版本操作，保证了返回数据的一致性；

4)子仓库与数据仓库在数据结构上的类似性，简化了开发和使用的复杂性。多级数据仓库为各级部门决策提供支持，可以依据各部门需要在子仓库上进行扩充，增加了系统的伸缩性。

5)OLAP 查询和视图更新的并行操作提高了数据仓库的更新效率和查询的实时性。

结语 本文对 Stanford 的 WHIPS 系统的数据仓库结构进行了改进，建立了多级数据仓库的体系结构，提出了维护多视图的双通道数据更新算法。多级数据仓库体系结构在企业中的建立为企业决策支持提供了有效的理论模式，针对改进结构的跨平台实现和与 Internet 的结合等问题尚处于讨论研究阶段。目前数据仓库和数据挖掘以及网络技术的发展，建立 B/S 模式的多级数据仓库将对大中型企业、科学研究、金融等众多领域带来不可估量的效益和进步，这些都为多级数据仓库的进一步扩充和发展提供了更广阔的前景。

参 考 文 献

1 Wiener J L,et al. A System Prototype for Warehouse View Maintenance. http://www.db.stanford.edu/warehousing/warehouse.html

2 Zhuge Y,et al. View Maintenance in a Warehouse Environment. In:Carey M J,Schneider D A,eds. Proc. of the acm sigmod conf. San Jose, California, May 1995. 316～327

3 Labrinidis A, Roussopoulos N. A Performance Evaluation of Online Warehouse Update Algorithms:[Tech. rep]. University of Maryland ,1998

4 赵玉源，梁阿磊，白英彩. 一种数据仓库的联机维护算法. 计算机工程，2000(8):78～79

5 焦容，陈金海，方方. 数据仓库多视图的并发控制分析. 小型微型计算机系统，2000(4)：407～409

6 李子木，李磊，徐明，等. 数据仓库的联机维护与下查. 计算机学报，1999(9)：988～992