

移动自适应更新机制

Mobile Adaptive Updating Mechanism

潘育飞 顾尔丹 吴震华 陈 纯

(CAD&CG 国家重点实验室 浙江大学计算机科学与工程学系 杭州 310027)

Abstract With the rapid development of mobile equipment and wireless communication, the Mobile Database Management becomes more popular. The main features of wireless communication such as low bandwidth and instability make the efficiency a critical problem in design and implementation of mobile database management. In the paper, we focus on the technique of updating the data in the mobile host. Based on the concepts of updating frequency, updating granularity, and adaptability, we set up a model on the dynamic decision of data updating operation.

Keywords Mobile database management system, Wireless communication, Adaptability, Mobile Agent

移动计算网络的逐步成熟和移动设备的日益普及刺激了移动环境下应用需求的迅速增长,而移动数据库是实现移动设备上应用的基础,因此,研究和开发移动数据库有着重大意义^[1]。在研究和开发移动数据库过程中,一方面由于移动计算环境具有无线通信开销高,带宽较低且不稳定,以及通信时延较大等特性,另一方面由于移动设备硬件的性能的提高,使得将操作所需的部分数据存储于移动设备变得必要。因此,目前大多数的移动数据库在移动设备端上的组件通常具备一定的数据存储管理和本地查询功能。一个相对独立且自治的移动主机,可以显著地降低网络通信的开销,并且获得更快的响应时间。

与此相对应,由于一个移动设备所需的数据子集往往来自多个分散的、异质的远程 DBMS,因此通常需要引入中间件^[2]将数据源和移动数据从物理上隔离开来,由中间件来维护数据源和移动端数据的同步和更新。中间件还可以实现对一些商业逻辑的封装^[3]。在移动设备上存在着许多如在设计商业决策或者危机处理等情况下发生的应用,它们对数据的实时性有着较高要求,使得中间件对移动端数据的更新操作将会比较频繁,因而更新操作性能的好坏对系统的综合性能会有很大影响。

本文主要针对数据的实时性要求较高的情况,讨论了一些影响移动更新操作性能的因素,并提出了一种新的可调更新频度和更新粒度的移动自适应更新机制。本文中首先简单介绍了一个我们已实现的可以在模拟环境下工作的移动数据库原型 ZMOBASE,并分析了该设计在移动端数据更新部分上的网络通信效率的不足。接着,针对这些问题提出了我们的改进方案:移动自适应更新机制。然后,给出了移动自适应更新机制较为详细的体系结构说明和性能分析。最后,我们提出了对今后发展方向的一些想法。

1 ZMOBASE 的原型系统

ZMOBASE 是我们自行开发的一个用于研究的移动数据库,由以下部分构成:

- 数据库服务器(DB Server)提供源数据库服务,一般采用各种成熟的商用 DBMS。

- 同步服务器(Synchronization Server)通过 MSS(Mobile Support Station)^[4],使移动数据库和源数据库保持适当的同步,其上运行的为移动数据库的中间件,主要提供工作数据子集的管理,以及在移动数据库和源数据库之间的上下复制/更新工作。

- 移动设备上驻留一个数据子集管理模块(MHDB),可以离线提供简单的数据库功能。

在设计并实现该原型后进行的性能模拟测试中,我们发现该原型在更新移动端的数据子集上存在一些缺陷,从而降低了系统的整体性能。

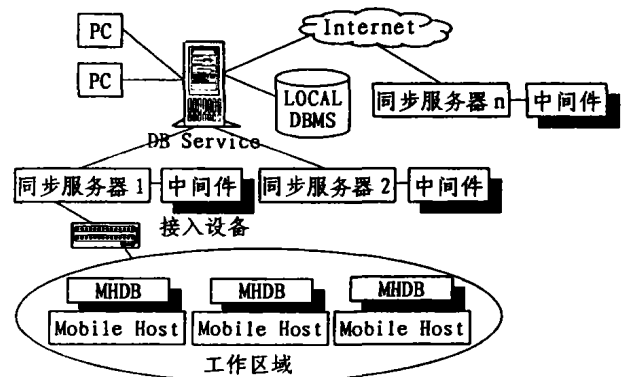


图 1 ZMOBASE 体系结构图

在原型中,对于同一移动设备的数据子集的不同数据项,我们采用相似的静态更新频率。但事实上,同一个数据子集中的不同数据项在不同的阶段使用频率有着很大的差别。对它们不加区别地采用同一更新频度,势必会造成对一些不经常使用的数据项的无谓更新,造成无线带宽的浪费,同时无法体现出热点数据所需要的高更新频率,于是造成了移动数据库系统性能和移动端应用性能的下降。

此外,原型中对更新数据组织的粒度太粗。由于原数据库的数据变动部分无法通过 ODBC 精确反映,原型只是简单地将发生变化的整个数据集传至移动端,而不是仅将必要的更改操作信息以及相关数据状态传至移动端,于是导致了更新中大量冗余数据的传输。当应用中需要相对频繁地更新移动

潘育飞 硕士研究生,主要研究方向为移动计算、人工智能、分布式计算。顾尔丹 硕士研究生,主要研究方向为移动计算、CAD/CG。吴震华 硕士研究生,主要研究方向为移动计算、人工智能。陈 纯 教授,博士生导师,主要研究方向为软件工程、CAD/CAM、人工智能、移动计算。

端数据以满足及时性要求时,系统的整体表现便会由于无谓的网络传输而明显下降。并且,这种简单的更新方式也会造成当发生网络带宽不稳定甚至断开的情况之后所继续的许多操作结果,无法在恢复连接后再次利用,虽然可能只是计算结果中的很小一部分受到源数据端数据更改的影响,但却必须在获得了有效的数据视图之后重新计算,一旦这些计算比较耗时,就会进一步使系统性能下降。

总之,由于原型没有充分考虑到待更新数据的应用特点和系统资源的合理利用,导致了整个系统无线带宽的极大浪费,同时也不能很好地满足移动端应用的要求。

2 自适应更新机制

针对前面对原型一些性能不足的讨论,我们提出了更新频度和更新粒度两个概念,通过选取合适的更新频度和更新粒度来获得较高的系统性能。更新频度是指当同步服务器感知到移动设备端上数据子集的一些数据项与源数据库中相应的数据项不再一致的情况下,对移动端数据子集进行更新的频度;更新粒度是指当一个多级组织的数据子集中的某一数据项发生变化后,为了更新其余数据拷贝,所选择的封装数据变动信息的级别。在关系数据库中,当某一数据集中的某一条记录中的一个字段发生了变化时,我们可以选择传输整个数据集,或传输一条记录,或仅传输发生变化的字段,和其他一些位置信息,这就是不同的更新粒度。

对于更新频度,我们希望在移动端应用可接受的性能范围内,通过降低当前一段时间较少使用数据项的更新频度来减少无线带宽的无谓占用,并使得热点数据占用相对较高的无线带宽,提高移动端应用的性能。

对于更新粒度,我们希望通过选择合适的更新粒度来节省宝贵的无线带宽。但是必须指出,这两个参数的选择除了考虑移动端应用性能和无线带宽之外,还必须考虑系统的其他资源如计算资源。更新粒度越细,一方面减少了无线带宽的浪费,另一方面却需要占用大量的计算资源,当计算的负荷过重时,系统的整体性能同样也会急剧下降。更新频度也有类似的问题,过低则会影响到移动端应用的性能降低。因而,对于各个移动设备上数据子集的更新频度和更新粒度的选取必须综合考虑到系统目前的各种资源的使用状况和移动端应用的要求,期望获得整体性能的最优。也就是说,我们希望系统能根据实际的运行状态,自适应^[5]地选择更新频度和更新粒度,进行高效低开销的更新。

下面给出更新机制的软件体系结构,并简单介绍各组成模块的功能。

2.1 体系结构

该机制在软件结构上分为两层,更新管理层位于同步服务器的中间件之内之上,更新实现层位于移动设备的 MHDB 中。更新管理层是两层机制中的关键部分,位于硬件体系结构中的同步服务器层,具有以下功能模块:

数据采集模块 根据需要和一定策略,通过 ODBC 接口与源数据相连,获得移动用户所需要的数据,该部分是更新管理层的一个最基本的功能。

变化定位模块 由于 ODBC 不能直接从数据源获得特定数据集发生变化的部分,因而在更新管理层中通过维护多个版本,比较新旧版本,定位变化发生的位置,并获得数据的变化量。然后根据更新粒度将变化的位置 and 变化量组织为向更新实现层提交的更新信息。

动态决策模块 正如我们在前面提及的,由于原型没有充分考虑到待更新数据的应用特点和系统资源的合理利用,导致了整个系统无线带宽的极大浪费,也不能很好地满足移动端应用的要求,因而我们引入一个动态决策模块对数据子集中不同数据项选择合适的更新粒度,同时,根据带宽资源和计算资源,自适应地调节更新粒度和更新频度。

通过选择合理的移动更新参数,我们期望移动数据库和运行在其上的应用的整体目标性能最优。

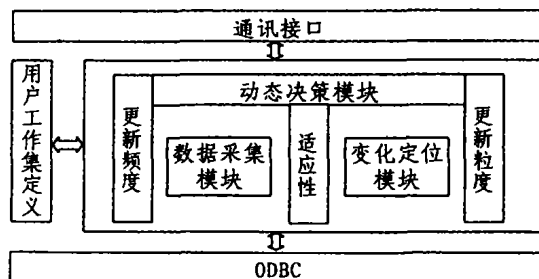


图2 更新管理层

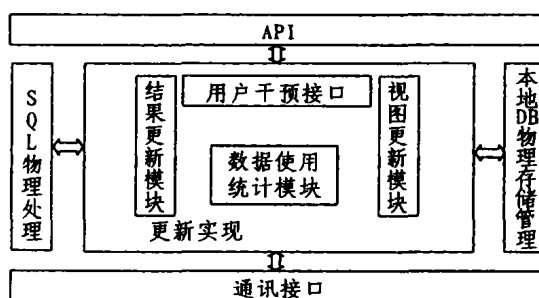


图3 更新实现层

更新实现层亦是二层机制必不可少的,它负责在移动端进行有关的更新决策信息的收集和实现移动端数据的更新,由以下功能模块组成:

视图更新模块 当更新实现层获得更新信息之后,通过相应的运算,修改原来的数据视图,以反映出数据源的变化;

结果更新模块 离线后重新连接后,将更新信息和需要更新的已操作结果进行计算,使结果反映出源数据的变化;

数据使用统计模块 统计各部分数据的使用情况,监视 MHDB 的活动情况,并向更新管理层提供这些信息,供更新管理层的动态决策使用。

实现中,由于对外接口未变,更新机制的变化对原型其余模块是透明的,即原型中其他模块可不作改动。

2.2 动态决策模型

鉴于动态决策的重要性,下面我们给出一个简单的模型来描述我们的决策系统。同时,为了充分尊重用户对某些特定应用特点的理解或对这些应用性能的特殊要求,我们同时支持高级用户对部分甚至全部数据项的策略定制,并且对该部分的数据项便不再进行动态决策。

目标函数: Overall Performance = $W_s * SystemPerformance + \sum W_{appi} * AppiPerformance$ 。

其中, W_s, W_{appi} 分别表示系统性能和各个移动设备应用的性能在整体性能评价中的权值。

环境变量包括:

数据项的使用频率 DUF(Data Utilizing Frequency): 反映移动端应用对移动端数据子集中各数据项的使用情况。

无线带宽状态 WB(Wireless Bandwidth):反映最近一段时间的无线通信(Wireless traffic)的拥挤程度。

同步服务器的计算负荷 CL(Computing Load):反映当前同步服务器上的计算资源的使用情况。

决策变量如下:

FUF(Fetch&Update Frequency)数据项的采集更新频率:该变量决定对所有数据项的采集更新频率,为各数据项采集更新频率的集合。

UTG(Update Transfer Granularity)数据变化部分的更新粒度:该分量决定将数据的变化控制在何种数据项类型中。

限制函数有:

$$\text{SystemPerformance} = W_b * \text{WBPerformance} + W_c * \text{CPPerformance}$$

$$\text{WBPerformance} = \text{WBP}(\text{FUF}, \text{UTG}, \text{WB}, \text{CL})$$

$$\text{WBP}(\text{FUF}_1, \text{UTG}) < \text{WBP}(\text{FUF}_2, \text{UTG})$$

当 $\text{FUF}_1 > \text{FUF}_2$

$$\text{WBP}(\text{FUF}, \text{UTG}_1) < \text{WBP}(\text{FUF}, \text{UTG}_2)$$

当 $\text{UTG}_1 > \text{UTG}_2$

$$\text{CPPerformance} = \text{CPP}(\text{FUF}, \text{UTG}, \text{WB}, \text{CL})$$

$$\text{CPP}(\text{FUF}, \text{UTG}_1) > \text{CPP}(\text{FUF}, \text{UTG}_2)$$

当 $\text{UTG}_1 > \text{UTG}_2$

$$\text{CPP}(\text{FUF}_1, \text{UTG}) > \text{CPP}(\text{FUF}_2, \text{UTG})$$

当 $\text{FUF}_1 > \text{FUF}_2$

$$\text{AppiPerformance} = \text{AP}(\text{FUF}, \text{UTG})$$

$$\text{AP}(\text{FUF}_1, \text{UTG}) > \text{AP}(\text{FUF}_2, \text{UTG})$$

当 $\text{FUF}_1 > \text{FUF}_2$

$$\text{AP}(\text{FUF}, \text{UTG}_1) > \text{AP}(\text{FUF}, \text{UTG}_2)$$

当 $\text{UTG}_1 > \text{UTG}_2$

$$\text{FUF} = \text{FUF}(\text{DUF})$$

$$\text{FUF}(\text{DUF}_1) > \text{FUF}(\text{DUF}_2)$$

当 $\text{DUF}_1 > \text{DUF}_2$

决策模块的功能是在当前的环境变量下,选择适当的更新频率和更新粒度,使得整体的系统性能最佳。模型中,简单地描述了各个函数随其变量变化的单调关系。在具体实现中,目前我们将各约束函数弱化为线性函数,并通过实验确定线性方程的参数。若想进一步获得更准确的决策,则还可以借助人工智能中关于决策系统的一些研究的成果。

但是,必须指出,决策模块的复杂度越高,系统的开销就越大。在我们的实现中,基本采用线性函数进行决策。

2.3 性能分析

和原型比较,性能的改进主要表现在:

第一,由于对移动端数据子集中的不同部分采用不同的

更新频率,一方面减少了对非热点数据不必要的更新次数,从而减少了无线传输;另一方面使得热点数据的更新更加及时,提高了移动端应用的性能。

第二,由于采用多粒度对移动端的数据项进行更新,在系统计算资源允许的情况下,通过较小的更新粒度,显著地减少了无线数据传输量。

当然所有这些性能的改进是以系统复杂性的增加以及同步服务器计算的增加为代价的,但由于在移动计算环境中,无线通信带宽异常宝贵,通过增加固定网络上的计算和通信换取无线通信效率的提高在一定的范围中是可以接受的。

同时由于我们的更新机制能够根据实际的系统运行信息和移动端数据的使用,对各项更新参数进行自适应的调整,因此整个系统可以较好地适应不断变化的状况。

结论和展望 本文通过引入更新频率、更新粒度的动态调节以及系统的自适应能力,使得对移动端数据子集的更新占用的无线带宽分配更加合理,整个系统期望得到的性能更加良好。但在实际应用中,同步服务器由于需要处理来自各个移动设备的大量的请求,有可能会形成整个系统的瓶颈,可在我们的改进了的二层更新机制上进一步将一些集中在同步服务器上的操作,如数据变化的定位和更新信息的组织分散到相应的源数据库,采用分布式的数据收集,降低对同步服务器的负担。比如可以运用渐以成熟的移动 Agent^[6]技术,通过派遣数据变化采集 Agent 至源数据库,采集并发回变化信息。

参考文献

- 1 Barbara D. Mobile Computing and Databases-A Survey. IEEE Transactions on Knowledge and Data Engineering, 1999, 11(1): 108~117
- 2 Pissinou N, et al. A Middleware-Based Architecture to Support Transparent Data Access by Mobile Users in Heterogeneous Environments. A research report in Center for Advanced Computer Studies & Center for Telecommunication Studies, University of Louisiana at Lafayette, 1998
- 3 Sandifer A, von Halle B. Business Rules: Capturing the Most Elusive. in High-Performance Web Databases Design, Development, and Deployment, Edited by Sanjiv Purba, 2001
- 4 Dunham M H, Helal A. Mobile Computing and Databases: anything new. SIGMOD Record, 1995, 24(4): 5~9
- 5 Noble B. System Support for Mobile, Adaptive Applications. IEEE Personal Computing Systems, 2000, 7(1): 44~49
- 6 Grabner M, et al. Agent technology: State of the Art: [Technical Report in Software Competence Center Hagenberg]. 2000

(上接第50页)

未来计算机应用是网络应用时代,特别是 Internet 这个全球范围内的公共通信通道,更是大家争相充分利用的宝贵资源。但既要借用此公共通信通道的方便性和经济性,又能保护公司私有的信息资源和商业秘密,存在相互矛盾的难点。解决这一问题设计 SecureGuard 的初衷。这个强有力的工具,具有巨大的市场应用潜力,其中三个具体应用简列于下:

·支持远程用户通过互联网访问内部网。具备访问公司内部私网(VPN)能力的机器,即安装了 SecureGuard 的机器,可以在任何地方接入互联网,可保密安全地访问公司的内部网络,安全程度不亚于在公司内部接入的方式。这给奔走于全球各地的公司移动员工(经理、董事长)提供了特别方便安全地与公司通信的解决方案。

·通过互联网连接公司内部的网络。如果有分布于各地的

不同分支机构,每个机构都有自己的局域网,只要各分支机构接入互联网的计算机安装了 SecureGuard,具有了 VPN 能力,则每个机构就可以通过 VPN 保密、安全地在互联网上通信。

·内部网的机器之间互连。利用 SecureGuard(VPN)还可以使物理连接在同一网络上的不同小组内部进行安全保密的通信。例如机构的不同部门之间互相保密,没有 VPN 能力的计算机即使连到网络上,也无法监听网络的数据传输。

参考文献

- 1 计算机技术开发人员宝典丛书编委会编. Sun Solaris 7 程序设计指南. 北京希望电子出版社, 2000. 6
- 2 IETF home page. <http://www.ietf.org>
- 3 蒋晓宇. 安全 IP 及公平交易协议的研究: [浙江大学博士学位论文]. 2000. 5
- 4 IP Filter home page. <http://coombs.anu.edu.au/~avalon/>