

基于操作日志的完井数据同步模型^①

吴雅娟, 任占广, 杜睿山

(东北石油大学 计算机与信息技术学院, 大庆 163318)

摘 要: 同步技术是提高移动数据库系统性能的一项关键技术. 依托完井移动平台, 结合完井业务数据, 提出一种基于操作日志的移动数据同步处理模型, 重点分析了该模型中的日志序列的生成以及冲突的消解, 并给出了具体的实现算法. 该模型允许移动终端在断接状态下对本地副本数据进行操作, 并通过移动事务日志记录用户行为, 当重新连接时, 根据服务器返回的日志序列进行同步处理, 使系统最终收敛于一致性状态. 同时还引入了日志合并、时间戳等方法, 有效减少了移动终端的资源消耗, 提高了同步效率. 实验表明该算法能快速高效地完成客户端与服务器之间的数据更新与交互.

关键词: 移动计算; 完井业务; 同步; 操作日志

Well Completion Business Data Synchronization Model Research Based on Operation Log

WU Ya-Juan, REN Zhan-Guang, DU Rui-Shan

(School of Computer & Information Technology, Northeast Petroleum University, Daqing 163318, China)

Abstract: Synchronization technology is one of the key techniques to promote the performance of mobile database systems. This paper which relied on the well completion mobile platform and integrated the well completion business data proposed a mobile data synchronization model which is based on operation log. In this model, the generation of log sequences, and the resolution of conflicts are discussed in detail, and the implementation algorithm is proposed. In this model, mobile devices are allowed to access the local copy data and locally commit the mobile transactions when the system is disconnected, meanwhile the user behavior can also be recorded in the mobile transaction log. According to the log sequence which is returned by the synchronous server, the mobile devices can update the local data when the system is reconnected to ensure the consistency of data. Besides, by introducing the merger of log and time-stamp into the model, the model can reduce the resource consumption, and improve the efficiency of the synchronization. The experiments proved that the algorithm can exchange the well completion business data between mobile devices and synchronization server rapidly and efficiently.

Key words: mobile computing; well completion business; synchronization algorithm; log sequence-based

完井是油气井建井工作最后一个重要环节, 直接关系到油气井的生产能力和经济效益, 同时又是油田开采的起点, 是从钻井开始到下套管、固井、射孔、下生产管柱的一项系统工程, 其业务流程主要包括方案设计、钻井、测试、射孔、作业、基础建设等多个环节^[1]. 在此过程中将产生大量的、多维的、分散的完井业务数据. 完井移动平台的建立, 实现了完井数据

的可视化管理以及完井业务的整合, 使工作人员可以更加方便的进行野外信息的采集, 不受时间或环境限制地进行完井业务的管理, 改变了以往的办公模式, 有很大的实际应用价值.

然而, 在实际应用中发现, 由于移动网络的不稳定、移动设备性能各异等原因, 客户端经常处于弱连接或断接状态, 无法与服务器进行信息的交换, 严重

^① 基金项目: 国家自然科学基金(61170132); 国家重大专项(2011ZX05020-007); 黑龙江省教育厅科学技术研究资助项目(12521055)

收稿时间: 2014-09-15; 收到修改稿时间: 2014-10-12

影响了平台的工作效率, 用户对其满意度不高. 考虑到移动计算环境的低带宽、长延迟、频繁断接和资源有限等特性, 移动端与服务器端持续的连接是不现实的. 为了不受外部环境的影响, 用户的大部分数据操作应该在本地进行, 但这又导致数据的短暂地不一致. 为了将本地的新数据更新到服务器, 同时服务器的新数据传递给客户端, 可以利用数据同步模型实现本地副本数据与中心数据库的交换, 使两端最终达到一致性状态. 该技术已经得到了广泛的应用. 在此基础上, 重点研究了如何提高数据同步和冲突消解的效率, 从而提高完井移动平台工作的流畅度.

目前, 研究人员已经提出了多种移动同步算法, 如两级同步算法^[2]、BAYOU 系统^[3]等. 前者需要在同步服务器上重做所有的移动事务, 大大增加了系统的负担; 而后者需要人为的干预来消解冲突, 严重影响了系统的适应性. 文献[4]提出了多版本控制算法, 该算法需要将数据的多个版本保存到同步服务器, 并采用特定的并发控制算法, 系统实现复杂度较高, 适应性和可扩展性较差. 为了克服目前移动同步模型的缺点, 本文参照“更新检测”和“同步协调”^[5]的基本思想, 以完井业务数据为基础, 提出了一种基于操作日志的移动数据同步处理模型. 该模型通过日志记录移动事务的操作轨迹, 同步服务器通过对比操作日志, 生成同步操作序列, 通过回放操作序列的方式实现数据的同步. 实验表明, 基于操作日志的移动数据同步算法与传统算法相比, 具有数据传输量小, 对同步服务器负载小等优点, 同时, 在移动完井平台应用中表现高效, 保证了完井业务数据的一致性和完整性.

1 方法原理

1.1 同步模型的基本原理

Jim Gray 给移动同步提出了四个目标, 即可用性与可伸缩性、移动性、可串行性、收敛性^[6]. 这四个特点决定了理想的同步系统不应受到网络环境的影响. 如图 1 所示的典型的移动同步系统^[7,8], 包括可信网络、移动节点、移动终端三部分.

可信网络通常由固定主机、同步服务器、数据库服务器组成, 为整个系统提供可靠的、实时的服务. 移动节点是可信网络与无线单元的桥梁, 通过它实现了二者信息的交互. 每一个无线网络单元内有多个移动终端, 每个移动终端对应一个副本数据库, 由本地的

移动数据库管理系统进行管理. 用户通过移动应用程序(如:完井移动平台)对本地副本数据进行操作, 从而导致了移动终端与移动终端之间;移动终端与数据库服务器之间;移动终端与固定主机之间数据短暂的不一致, 需要通过同步机制实现各个单元的数据交换, 保证数据的完整性和一致性. 由此得到了如图 2 所示的同步系统模型.

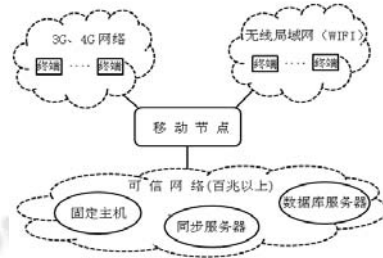


图 1 系统总体框图

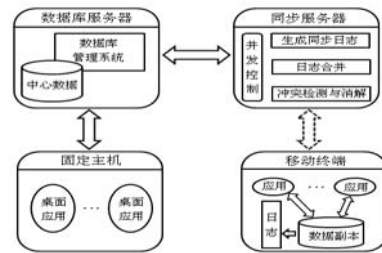


图 2 基于操作日志的移动同步系统模型

由图 2 所示, 操作日志随着移动终端通过同步服务器与数据库服务器的信息交换循环的发生状态的转换^[9]. 即空状态、记录状态、同步状态. 如图 3 所示.

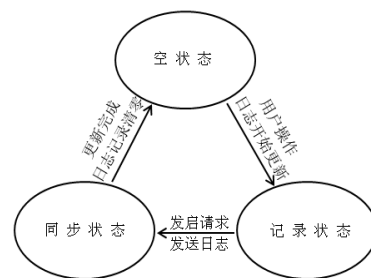


图 3 操作日志状态转换

(1)空状态

移动终端与同步服务器进行同步处理后, 双方分别根据操作序列更新了本地数据. 在某种程度可以说两者的数据是相同的. 此时移动事务操作日志没有一条记录, 处于空状态.

(2)记录状态

在断接的情况下,用户通过移动终端只能对本地数据副本进行“实时”操作.同时操作日志自动记录用户的操作轨迹,如数据项,数据对象的变化状态等等,并且每一条记录都加盖时间戳.操作日志的体量随着用户的操作不断的增大,此时,操作日志就由空状态转换到了记录状态.

(3)同步状态

同步可以由用户手动开启,也可以通过网络监听模块(当检测到移动终端处于 WIFI 状态下,开启同步,完成数据更新.从而降低通信费用,缩短同步时间)自动开启.首先移动终端将操作日志发送给同步服务器;随后同步服务器对操作日志进行冲突检测、日志合并、生成同步结果集等操作;最后,移动终端更新本地数据副本,从而与数据库服务器中心数据达成一致.整个过程中,移动终端始终处于同步状态.同步处理过程结束后,操作日志被清除,恢复到空状态.

1.2 冲突消解

数据冲突是同步系统比较常见的现象,冲突的消解通常有特权法、协调法、随机法、时间顺序法^[10-13].本文在时间顺序法的基础上,结合操作日志提出了一种时间戳方法.该方法在记录移动事务的变化时,同步加盖时间戳来标记事务发生的具体时刻.当多个事务对一个数据对象进行写操作时,通过对比时间戳,以最近时刻时间戳为准,进行写操作.通过该方法可以快速实现事务日志的冲突消解,同时通过对带有时间戳的事务日志进行分析,可以方便地得到用户的操作频率、访问次数、在线时间等,通过对这些信息的分析,可以获知用户经常访问数据的类型,操作的习惯等,可以利用移动推送机制提示用户可能关心的信

息的最新动态,提高系统的友好度和智能化^[14].一旦出现时间戳相等的极端情况(只能在服务器端发生,在客户端不可能发生),那么将舍弃掉该条日志,保持中心数据不变,并友情提示用户.具体流程如图 4 所示.

2 同步模型的算法实现

2.1 移动事务日志

传统的事务日志除记录包括插入、更新、删除的基本操作外,同时还记录提交、回退等事务模式的变化,占用空间较大.如果采用传统的方式完成移动事务日志的记录,将会使移动终端有限的空间变得更加紧张.所以将采用简化的移动事务日志结构.即通过日志记录和日志合并两种方法缩小日志占用的空间.在日志记录方面,仅记录插入、更新、删除操作并且对这些操作日志加盖时间戳.对于更新操作,需要记录更新前后数据对象的变化.对于删除和插入操作,需要将完整的删除记录和插入记录转存到操作日志中.在日志合并方面,根据日志合并的规则,设计日志合并算法.操作日志每记录一条数据都调用日志合并算法,与日志中现存的记录进行冲突检测,根据时间戳,仅保留产生冲突的最新记录.如表 1 所示的移动事务调度情况(其中 MT1,MT2,MT3 为移动终端上执行的三个移动事务);表 2 是传统的事务日志;表 3 是采用合并算法的移动事务日志.

表 1 移动事务调度表

SHCHEDULE	MT1	MT2	MT3
R1[A]=a1	R1[A]=a1		
R1[B]=b1	R1[B]=b1		
W2[A]=a2		W2[A]=a2	
R3[A]=a3			R3[A]=a3
W3[B]=b2			W3[B]=b2
R1=c1	R1=c1		
W1[B]=b3	W1[B]=b1		
W3[A]=a4			W3[A]=a4
W2=c2		W2=c2	
W3=c3			W3=c3

表 2 传统事务日志表

操作方式	数据对象	结果集
Read	A	a1
Read	B	b1
Write	A	a2
Read	A	a3
Write	B	b2

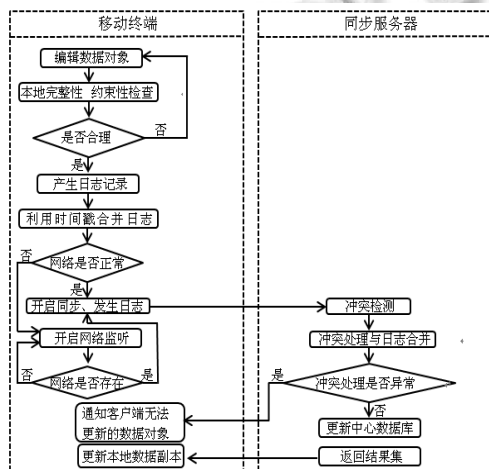


图 4 操作日志冲突检测与消解流程图

Read	C	c1
Write	B	b3
Write	A	a4
Write	C	c2
Write	C	c3

表 3 利用合并算法的事务日志表

操作方式	数据对象	结果集
Write	B	b3
Write	A	a4
Write	C	c3

2.2 移动同步算法

基于操作日志的同步算法包括日志上传和数据更新两个阶段。第一阶段是移动终端将操作日志传递给同步服务器，同步服务器进行日志的冲突检测和消解，更新中心数据库并生成移动事务操作序列。第二阶段是移动终端根据同步服务器回传的操作序列，对本地数据进行更新，以达到本地副本数据与中心数据的一致目的。算法 1 所示的日志上传流程，包括日志的传递、冲突的检测、操作序列的生成三部分。算法 2 为数据更新流程。

算法 1: 日志上传。

INPUT: Send Log

```

1): IF MTLog ∈ ( DeleteLog、WriteLog、InsertLog)
then
2): MTLog insert CommitLogMT;
   //冲突检测与日志合并
3): Lock All;
4): For i=0 to CommitLogMT do
5):   For j=0 to CommitLog[i] do
6):     IF CommitLog[i] ∩ CommitLog then
       //对比时间戳, 消解冲突
7):       If CommitLog[i].time > CommitLog then
8):         CommitLog[i] put TempLog ;
9):       Else
10):        CommitLog put TempLog;
11):      Endif
12):     Else
13):       TempLog put  UpdataLog;
14):     Endif
15):   Endfor;
16): Endfor;
17): Unlock All;

```

```

18): For All UpdataLog;
19): IF UpdataLogi ∈ TransLog;
   //更新中心数据库
20):   Exe  UpdataLogi
   //生成返回序列
21):   UpdataLogi put RrturnSeriLog;
22): Else
23):   Continue;
24): Endif;
25): Endfor;
OUTPUT: Return SeriLog
算法 2: 数据更新
INPUT: StopLog; //日志停止更新
LOGdown ; //下载更新日志, 准备更新本地副本
1): LOGdown Convert MLOG;
2): For All MLOG ∈ MCommitLog
3):   DownDataSet;
4): Endfor;
5): Lock;
6): For All X ∈ DownDataSet;
7):   If Xtime > Xlogtime
   //更新本地数据
8):     Exe MLOG(x);
9):   Else
10):    Continue;
11):  Endif;
12): Endfor;
13): Unlock;
OUTPUT: 提示客户端更新完成

```

3 实验

本文采用大庆油田完井移动平台移动事务日志进行同步更新实验，依照 3 节中的方法，通过对移动事务的记录与合并生产上传日志(如表 3)，使同步服务器对客户端上传的操作日志进行冲突消解、数据写入、返回移动更新序列(如算法 1)，客户端利用回放日志序列完成本地数据副本的更新(如算法 2)。为验证本文提出的同步模型的性能，以移动完井平台和完井服务器为依托，分别考察了该模型的处理效率和并发控制。

利用 Java 语言实现了该算法，如图 5 所示，假设仅有一台移动终端，操作日志分别为 100、200、300、

400、500、600、700、800、900、1000 的情况下,逐一的向同步服务器发送同步请求的响应时间(处理时间)。如图 6 所示的 10 台移动终端(日志数目都为 500 条)并发向同步服务器发送同步请求的响应时间。

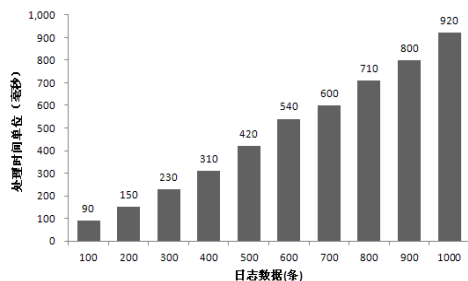


图 5 完井移动事务日志体量与处理时间的关系

实验结果如图 5 所示,可以看出在操作日志在 1000 条以内时,同步服务器处理时间与日志体量成基本线性关系,平均处理时间为 500 毫秒左右,日志条数为 1000 时,同步服务器操作日志的处理时间是 920ms。

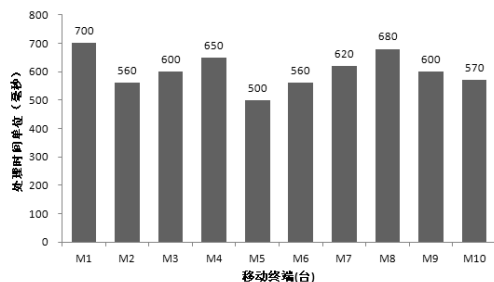


图 6 移动终端并发访问量与处理时间的关系

实验结果如图 6 所示:可以看出当日志体量在 500 以内、多个客户端并发同步时,同步服务器的处理时间(系统的响应时间)在 1000 毫秒与 500 毫秒之间,平均的处理时间在 600 毫秒左右,并发处理时间在 100 毫秒左右。

综上所述,基于操作日志的同步模型在同步处理、并发控制上都保持秒级以下的处理时间,表现高效。

4 结语

本文依托移动完井平台,提出了一种基于操作日志的完井数据同步模型,用于实现本地数据副本与中心数据的同步。实验证明,无论是多用户的并发访问,还是大数据量的用户请求,同步服务器都能及时快速的完成数据同步。接下来的工作,一方面要对网络监

听模块进行智能化处理,实现在充电、WIFI 等理想情况下,启动同步模块完成同步处理,降低对移动终端的负载和通信费用;另一方面要通过差值算法提高日志冲突的检测速度,提高同步的效率。

参考文献

- 1 齐奉忠,申瑞臣,刘英,等.国内固井技术现状问题及研究方向建议.钻采工艺,2004,27(2):7-10.
- 2 Gray J, Helland P, O'Neil PE, Shasha D. The dangers of replication and a solution. Proc. of the 1996 ACM SIGMOD International Conference on Management of Data. Montreal, Canada. 1996. 173-182.
- 3 Demers A, Petersen K, Spreitzer M. et al. The bayou architecture: Support for data sharing among mobile users. Proc. of the IEEE Workshop on Mobile Computing Systems and Applications. Santa Cruz, CA, USA. 1994. 2-7.
- 4 Phatak SH, Badrinath BR. Multiversion reconciliation for mobile databases. Proc. of the 15th International Conference on Data Engineering. Sydney, Australia. 1999. 582-589.
- 5 Gupta D, Sagar K. Remote file synchronization single-round algorithms. International Journal of Computer Applications, 2010, 4(1): 32.
- 6 Gray J, Helland P, O'Neil PE, Shasha D. The dangers of replication and a solution. Proc. of the 1996 ACM SIGMOD International Conference on Management of Data. Montreal, Canada. 1996. 173-182.
- 7 王若莹,李梁,张润洲,王建.一种移动数据同步算法.计算机技术与发展,2010,20(12):137-139.
- 8 丁治明,王珊,孟小峰.移动复制数据库系统冲突检测及消解策略.计算机学报,2002,25(3):298-299.
- 9 张晓杰,刘杰,马志柔,叶丹,高洪涛.基于操作日志的云存储服务多终端同步算法.计算机工程与设计,2013,34(11):3896-3897.
- 10 黄东.基于 sqlite 的移动嵌入式数据库同步系统的研究和开发[硕士学位论文].武汉:华中师范大学,2009.
- 11 陈舒,姜宁康.基于多方面完整性检测的移动数据库同步机制.计算机应用,2009,29(6):184-187.
- 12 陈历胜,郭海滨,叶飞跃.移动计算环境下的一种同步复制模型.计算机应用,2008,28(10):2544-2546.
- 13 魏凯斌,崔兆顺.移动数据库同步机制的分析与研究.自动化与仪器仪表,2011,2:26-27.
- 14 施滔滔,马永征.移动科研在线中基于冲突解决的数据同步策略.计算机工程,2014,40(20):119-122.