第8卷 第4期 华侨大学学报自然科学版 Vol. 8 No. 4
JOURNAL OF HUACHIAO UNIVERSITY
1987年11月 (NATURAL SCIENCE) Nov. 1987

Dimension多用户系统的改进方案

谢 红 毛

〔计算机科学(电脑)系〕

摘 要

本文根据笔者参加应用美国North Star 公司的 Dimension 系统开发应用软件的实践，分析了 Dimension 系统作为多用户开发环境所存在的问题——在未进一步开发前，无法真正支持多用户应用.文中就如何开发以适应于多用户的使用提出了解决的方案和技术,并经实践认为可行.

一、原支持环境与多用户系统开发环境的差距

Dimension 系统力图作为一种与 IBM PC/XT 兼容的多用户／多处理器微型机系统，是采用多处理器的形式，把多处理器组成一个协作系统，並以主机核心模块、MS-DOS 操作系统和dBASE-Ⅱ数据库管理系统作为其主要软件支持。该系统原支撑环境已基本上实现了对各处理器的任务分配，这是基于下面事实保证的：中央板和各工作站各自配有处理机并拥有自己的内存，中央板上的处理器作为运行实时多任务的核心，各工作站处理器独立工作，当它们需要系统资源时，工作站通过中央板的I/O口请求资源，当中央板处理器准备就绪响应请求时，借助于I/O口与工作站处理器建立的紧耦合关系，通过两个处理器可以访问到的内存区进行数据传送工作。当完成数据传送时，中央处理器再次使用I/O口使工作站处理器独立工作。但是作为多用户系统环境的另一个要解决的重要问题即共享资源的管理,Dimension系统并没有解决。该系统拥有单个硬盘，可以由各工作站所共享，而系统并没有提供对硬盘这一重要共享资源的完善管理，加上 dBASE-Ⅱ 的不完备性，产生如下问题。

1. MS-DOS 硬盘共享区中空间管理混乱

Dimension 系统将硬盘分成三种不同类型的盘区即私用区、公用区和共享区。每个用户至少要有一个私用区，私用区存贮只有用户本人才能使用的文件，而公用区通常存贮文件程序，公用区中的文件只能读，不能修改和删除，共享区用来存放多个用户共享的数据文件，可以为各工作站用户共享。由于文件空间的申请、释放由各个工作站分开独立管理，彼此没有通讯互不了解，因此会引起盘区空间管理的混乱，比如两个工作站上的用户各自在硬盘共享区中申请一个文件空间，由于互不了解，可能申请到同一空间，导致重叠。

2. 用户数据文件修改不一致性

本文1987年2月18日收到.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

由于dBASE-Ⅱ没有提供多用户情况下数据一致性的保证措施，不同工作站同时对同一共享数据文件进行修改会产生复盖现象，以后入为主，即用户甲、乙同时对某一数据文件修改，结果用户甲修改过的数据又被用户乙修改，以致引起用户甲所修改的数据丢失。

3．dBASE-Ⅱ存在的不完备性

dBASE-Ⅱ中有的操作命令或语句的语义是不确切的，比如Total命令中filename的数据结构说明，Do while命令中循环控制条件的设置，Delete命令中scope中 n 的选定说明，find命令中 n 个字段变量组合的关键字串的说明等都有不确切的地方，有的说明隐含着错误，如何正确使用这些命令，本文不详细阐述，本文着重对修改命令Replace加以较详细的说明。该操作命令是修改数据库经常使用的重要手段，在打开索引的情况下，利用该命令修改数据库会产生出错。我们对该命令使用作了测试，分析产生出错的原因，并提出了解决的办法，关于其解决办法在下面详述，这里仅就检测结果说明如下：

(1) 按记录号顺序修改且关键字各位相同：当关键字长度为5，测试500个记录以内文件，第一个出错的记录是478*，当关键字长度为20，测试300个记录以内文件，第一个出错的记录是107*，当关键字长度为30，测试200个记录以内的文件，第一个出错的记录是59*。

(2) 按记录号顺序修改，且关键字值随机生成：测试500个记录的文件，当关键字长度分别为5，20，30时，相应的第一个出错的记录分别为494*，110*，56*。

(3) 记录号随机生成，且关键字各位相同：测试500个记录的文件，当关键字长度为5和20时，无出错；而当关键字长度为30时，第一个出错的记录为214*。

(4) 记录号随机生成，且关键字值随机生成，情况与(3)基本相同。

之所以会产生上述的情况，是由于dBASE-Ⅱ的索引结构采用B*树，Replace对索引的修改是由删除和插入两个动作完成的，当删除产生节点空而插入又需要分裂，就产生索引文件混乱，所以对索引不允许修改。

此外，dBASE-Ⅱ不完备性除了表现在某些命令、语句不确切外，还表现在有些命令虽然语义是确切的，但运行效率很低，比如Jion、Sort等命令，这个问题可以通过用其它命令组合来替代，以提高效率。

二、解决的技术

根据上面提出的一些问题，结合我们在开发应用软件的实践，提出了如下一些技术。

1．对盘共享区使用的限制

对盘数据共享区中的文件，将其分成两类，即定长文件和可定长文件。前一类文件长度已确定，在运行中没有申请、释放空间的问题，这部分盘区管理就不会产生混乱。对于后一类文件，在初始化时根据事前的估计和预测分配足够长的空间，以确保在一段时间内足够用。对该类文件修改就用修改命令进行，增加记录通过先找空记录，再用修改命令置进有关信息来代替，文件长度并不增生，删除记录通过用修改命令把记录置空来代替。对这类文件要进行定期地复制、刷新，以便保存信息，同时回收这部分空间，使其可被再使用。总的来说，对硬盘数据共享区，每个工作站不允许有改变盘图空间的操作，在运行中共享文件不允许增生、删减。诚然，这种办法不是一种彻底的解决办法，彻底的办法应该改造MS-DOS的

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

文件管理。根据我们的实践，采取这种灵活的办法是有效可行的。应当注意，为了避免引起盘区管理混乱，在用户调试程序时不要在共享区里进行，而应在各自工作站私用区进行，当调试成功后，再由私用区倒入公用区。

2. 数据共享一致性(数据丢失)的保证

如上所述，Dimension 系统中的 dBASE-Ⅱ 并不能保证在多用户情况下的数据共享，各个工作站对文件的修改是串行从盘上读出，对文件的修改是在各个工作站的缓冲区进行，修改完毕再串行写回盘，所以各个工作站对共享数据文件的修改实际上是并行进行的。这样会产生覆盖现象，即甲、乙两个用户在各自工作站上同时修改某个共享文件，结果文件以两个用户之中最后一个写回硬盘的文件为准，而前面一个写回硬盘的信息被覆盖了，变为无效，即“后入为主”。

通过在操作系统一级上增加一对可在编程一级上调用的 P、V 操作，即在工作站之间建立一种简单的低级通讯，使各工作站对盘区共享数据的操作成为串行，即使两个不同用户对同一共享文件的读盘、修改、写盘整个过程成为完全的串行，这样就不会产生上述的“覆盖”现象，可保证共享数据的一致性。

具体 P、V 操作主要过程如下：

P 操作：(a)测试共享资源(数据文件)是否被占用；(b)若被占用则等待释放，否则加占用标志；(c)读出文件。

V 操作：(a)将缓冲区内容(已修改过)写回硬盘，即排出缓冲区；(b)撤除占用标志。

3. 解决 dBASE-Ⅱ 的不完备性

如何解决 dBASE-Ⅱ 的不完备性，本文仅就怎样使用 Replace 命令，正确地修改文件提出解决的办法，即引入前后台工作方式。

将 Dimension 系统中的工作站设置一台作为监视工作站即后台工作站，再设置若干台工作站作为前台工作站，前后台工作方式示意图如图 1 所示。

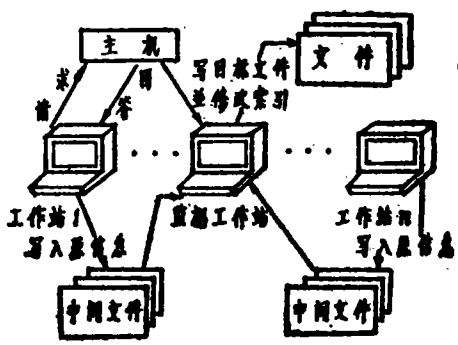

图 1　前后台工作方式

将共享文件分为关键性文件和非关键性文件。关键性文件是指在具体应用时对数据操作实时性要求高、索引不必要修改的文件，而其余文件称为非关键性文件，这类文件索引允许修改。比如在我们研制的分布式酒店管理系统中，房态表文件设置有房号索引、级别索引等，由于房号、级别都是固定的，所以不必修改这两个相应的索引文件，将房态表文件称之为关键性文件。又比如旅客住宿登记表设置有姓名、来住日期等索引，而姓名和来住日期随着旅客不同随时变动，必需不断地修改相对应的索引文件，旅客住宿登记表文件被称为非关键性文件。

对于关键性文件，其修改由前台按通常对文件一般的操作进行，采取即时处理，由于其关键字值不变，所以没有修改索引文件问题，因此不会产生错误。

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

对于非关键性文件采用前后台处理方式，把前台工作站要修改的共享文件称之为主文件，同时在每个前台工作站设置一个中间文件，其结构包含文件名、记录号、信息串，其中信息串的长度为诸主文件体积的最大值，中间文件初态时为空，它可以由各前台工作站与监视工作站共享，但其它前台工作站不可共享。

非关键性文件的修改过程大概是：在前台工作站，根据已知的主文件的数据结构，将要修改的主文件记录的全部信息转换并组装成一个字符串，并按中间文件结构，用 Replace 命令产生相应中间文件记录值，在产生中间文件记录时要通过加载 P、V 操作，以保证数据的一致性，一旦中间文件形成之后再通过监视工作站倒入原主文件。其做法是：根据主文件字典将中间文件的信息串还原成主文件记录值，随后用 Replace 命令在不打开索引的情况下对主文件的相应记录进行修改。

监视工作站主要任务是把修改过的中间文件倒入原主文件，并且定期对修改过的文件重建索引，其工作大致过程如图 2 所示。

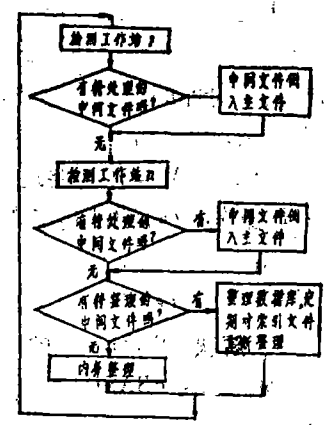

图 2 后台工作站控制过程

这里要指出一点，为什么我们要采用上面所述的前后台处理方式，而不采用如下比较简单的处理流程，即对共享主文件不打开索引，在各工作站上直接由 Replace 命令进行修改，然后由后台随机地对主文件进行索引重整。这主要是考虑到系统的实时响应问题。如果采用后者的简单处理流程，那末，对于多用户系统可能出现多个工作站，同时要对同一个共享文件进行修改，为了保证数据的一致性，在进行修改时，需要通过 P、V 操作，使修改成为串行等待。假设 m 为文件共享的工作站数，T 为工作站P、V操作及写文件（修改文件）的时间，那末在运行中某一工作站执行对共享文件的修改平均时间为 $mT/2$，最极端情况需要 $(m+1)T$ 操作时间，如果有 4 个工作站对某一文件进行共享，T 为 6s(实测时间)，那末系统在运行的最坏可能是一个写操作就须花 30s 的时间，这是不能满足用户界面的实时性要求的。

采用前后台处理方式的优点是，解决了Replace命令在打开索引情况下产生的修改错误。因为在这种方式处理中，使用Replace命令均不打开索引，避免了出错，而修改过的文件又由后台定期建立索引，重新整理数据，在整索引时基本上不影响前台工作，保证查询实时性。关键字长度为20个字符，长度分别为100个记录、500个记录、1000个记录的索引文件，重整时间分别为 6s、35s、70s。这里虽然在文件修改时同样需要 P、V 操作，但是由于只需要每一操作工作站与监视工作站两两相关，等待时间只是一个工作站形成中间文件的时间。这种方式不足之处是需要专门设置一个工作站作为监视工作站，增加了系统开销。分布式酒店管理系统就是采用这种前后台处理方式的。

三、实　　例

我们研制的分布式酒店管理系统就是在美国 North star 的 Dimension 系统支撑环境下

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

开发的，采用上述技术解决了研制过程中所遇到的一系列技术问题。

1. 共享数据文件的设置

一个饭店最多可以接纳多少旅客以及共有多少房、床这是确定的，为此将旅客住宿登记表以及房态表作为定长文件，而对于旅客预定表、电话单、服务费用单、房租单、帐单、离店旅客表等这些文件事先无法确定其长度，根据饭店实际情况，在系统初始化时确定各个文件的足够长度，比如电话单取20000个记录等。即把这类文件设置为可定义的定长文件。

2. 引入P、V操作

用dBASE-Ⅱ命令和汇编命令编制二个标准命令文件即P操作和V操作命令文件。解决数据共享一致性(数据丢失)问题。

3. 系统的布局

该管理系统由总服务台子系统、财务子系统、经理室子系统、餐厅子系统、商场子系统、车队子系统等组成。总服务台子系统配有4个工作站作为前台工作站，并专门设置一个工作站作为监视后台工作站，其它子系统分别配有一个工作站，即一般工作站不涉及前后台处理。在系统开工之后，各个子系统在各自的工作站上注册，独立处理各个部门的工作。总服务台子系统是整个酒店管理系统的主要部分，借助于监视工作站采用前后台的工作方式。该子系统负责房务处理、帐务处理、预定处理、查询处理和内部事务处理，这五种处理功能可以根据服务的需要在总台所配置的4个工作站中的任一个工作站上并行地执行。一般前台处理一个旅客入住登记只需40s左右，处理一个旅客的结帐只需1.5 min，其中包括人机交互过程。如果包括由打印机打印详细帐单也只需2 min多钟。通常情况下，总台上的4个工作站可以分别同时处理预订、入住、结帐、查询多种服务。如果出现旅客入住峰期，4个工作站可以同时用于做入住登记、房务处理，如果出现旅客离店峰期，4个工作站可以同时用于作帐务处理。在这两种情况下，都能满足用户界面的实时性要求。

分布式酒店管理系统已通过鉴定，专家们对该管理系统评价较高，充分地肯定了该系统所采用的技术。

四、结　束　语

就笔者所知，目前美国North star公司推出的Dimension系统在我国各行各业已拥有一批用户。本文就如何在Dimension系统支撑环境中开发，以适应于多用户的应用提出了解决方案和技术，其思想和实现方法可提供同行借鉴。

参　考　文　献

〔1〕谢红毛、陈勤勤、余坚、陈维斌，分布式酒店管理系统，华侨大学学报(自然科学版)，8,3 (1987)。

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

Effective Measures for the Modification of Dimension Multi-user System

Xie Hongmao

Abstract

Based on experiences obtained from developing application software for Dimension System put forward by U. S. North Star company, this paper analyses some serioua problems occurred so long as this system is used as multi-user developing environment. Actually, this system is incapable of supporting multi-user application unless it is further developed.

For solving these problems, some effective measures including scheme and technic are offered here. They are proved by practice to be feasible in developing multi-user applications.

© 1994-2010 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net