

# Oracle 数据库性能调优技术与实现

郭海峰<sup>1,2</sup>, 阳国贵<sup>2</sup>

(1. 湖南省气象科技开发中心, 长沙 410007; 2. 国防科技大学计算机学院软件所, 长沙 410073)

**摘要:** 讨论了麒麟操作系统上 Oracle 数据库系统性能优化与调整技术, 以 TPC-C 测试程序作为手段, 对不同的内存分配、数据库进程结构、物理数据库设计、等待事件的观测与处理方法进行了分析。实验结果表明, 采用合理的性能优化措施, 可以将 Oracle 数据库系统的性能至少提高 50%。

**关键词:** 麒麟操作系统; 性能优化; TPC-C 测试

## Skill and Implementation for the Oracle Performance Tuning

GUO Haifeng<sup>1,2</sup>, YANG Guogui<sup>2</sup>

(1. Hunan Meteorological Science and Technology Development Center, Changsha 410007;

2. Institute of Software, College of Computer, National University of Defense Technology, Changsha 410073)

**【Abstract】** This paper discusses the skill and implementation for the Oracle DBMS on the KYLIN OS, using TPC-C benchmark as the instrument, and it analyzes the memory's distribution of the system, the structure of Oracle's process, the physical design of the database, the observation and the treatment of the wait events. As the result of the experiment, the performance of Oracle database can be improved by 50% at least, by using reasonable tuning skill.

**【Key words】** KYLIN OS; Performance tuning; TPC-C benchmark

### 1 概述

银河麒麟操作系统是具有自主知识产权的服务器操作系统, 具有安全、高性能、高可用与 Linux 二进制兼容等特性, 可以支持各种应用软件、中间件、数据库系统的稳定运行, 支持各种信息平台的搭建。Oracle 是目前最流行的几个数据库管理系统之一, 拥有较大的市场占有率和众多的高端用户, 成为大型数据库应用系统的首选后台数据库系统。Oracle9i 是业界第一个完整、简单的用于互联网的新一代智能化的、协作各种应用的软件基础架构。

Oracle 数据库系统中提供了系统监控的各种跟踪信息, 警告信息, 系统动态视图, 可以调整的系统参数, 性能调整软件包等手段和方法, 但由于它是一个复杂的基础软件, 需要对系统中的各种性能症状有系统而全面的了解, 才能进行系统的性能调整和优化。而内存分配、数据库进程结构、物理数据库设计、各种等待事件的观测与处理方法是系统性能调整的主要方面, 将在本文进行深入分析和讨论。

TPC-C 是国际事务处理委员会制定的针对 OLTP 的性能测试标准。TPC-C 模拟了一个库存、销售系统, 该公司由分布在不同地区的仓库和地区分公司组成, 当公司业务扩大时, 将建立新的仓库和分公司。标准中规定每个地区分公司服务 3 000 名客户, 公司总共拥有 100 000 种商品, 分别存储在各个仓库中。图 1 给出了公司、仓库、分公司、客户的层次结构关系。

为了评价服务器平台、数据库管理系统的在线事务处理性能, 通常采用 TPC-C 标准作为一个客观的评价尺度, 以每分钟处理的新订单事务数量 (TPMC) 作为该环境下事务处理的性能评判标准, 这样对系统调整的目标就是力求使系

统的 TPMC 达到最大。

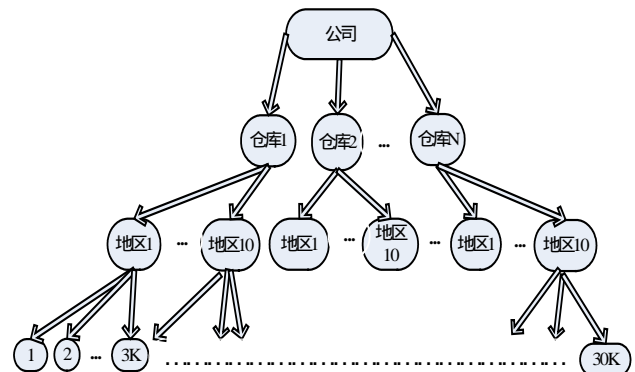


图 1 TPC-C 模型客体关系图

### 2 测试环境

为了研究麒麟操作系统上 Oracle 数据库系统的性能调优技术, 建立如图 2 的测试环境。

测试环境采用联想万全 R530 作为测试服务器, 硬件配置为 :2\*Xeon 2.4GHz CPU + 2\* 37GHz Ultra 320 SCSI 硬盘 + G2B 内存+双千兆网卡。在麒麟操作系统 V2.0 运行 Oracle9i。测试终端为联想台式机和 IBM 笔记本。当一台前端机器不够时 (如其 CPU 达到 95% 以上), 就配置多台前端机。当用两台以上的测试前端机时, 需要将其中一台配置成主测试前端机, 其余为从测试机。通过 TPC-C 测试程序在 Oracle 数据库中建立需要的相应关系表, 并根据上述机器的能力配置加载

**作者简介:** 郭海峰(1975 - ), 男, 硕士生, 主研方向: 数据库系统技术及应用; 阳国贵, 副教授

**收稿日期:** 2006-01-12 **E-mail:** 13908475447@hnmcc.com

30 个仓库数目的数据以后，开始测试程序的运行。观察 30min~40min 的测试运行，检查各事务处理的平均时间，看是否达到标准规定的时间范围。如果在规定时间范围内，就可以继续增加仓库数目。随仓库数目的增加，数据库规模增大，系统中的事务处理时间也会随之变化，这样需要观察服务器和 Oracle 数据库的表现，并进行配置的改变和优化，使得可以支持的仓库数目达到最大。

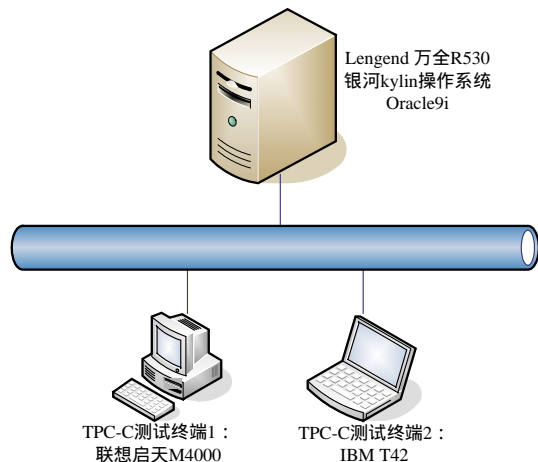


图 2 系统应用测试环境

### 3 性能调优技术与实现

对麒麟操作系统上的数据库进行性能调整，有两种基本的方法：自上而下的调整方法和基于性能调整准则的方法，前者是在检查数据库内存结构、物理 I/O 所相关的调整问题之前，先进行应用程序设计与 SQL 的调整。该方法假设不好的 SQL 语句和应用程序设计是造成低性能的关键。我们先对 TPC-C 中的事务类型操作以及相关的物理数据库的设计进行分析，观察对系统性能影响最大的 SQL 语句的性能调整。后者将重点放在找出与解决调整问题的较一般技巧上，比如如何检查系统性能，收集 Oracle 统计数据，将已确认的问题与已知的性能问题进行对比分析，并进行相应调整，并确定是否达到调整目标，如没有继续上述过程，直到满意为止。该方法要求能对系统的统计信息进行解读，并采用相应的调整措施。

系统健康状态的信息来源主要有系统的报警日志信息（Alter Log），在麒麟操作系统中，遵循 alter\_SID.log 的命名规定。后台跟踪文件信息，位于 init.ora 文件中参数 BACKGROUND\_DUMP\_DEST 指定的位置，由进程名加编号组成，如由 Dbw0\_nnnn.trc、Lgwr\_nnnn.trc 等文件组成。用户跟踪文件信息，位于 init.ora 文件中参数 USER\_DUMP\_DEST 指定的位置，文件名中包括了用户进程标识符。只有设置 init.ora 中的参数 SQL\_TRACE=TRUE，以该实例为背景的进程都将创建它们自己的跟踪文件，这种情况下要注意信息的过度增长。通过 Alter session set SQL\_TRACE=TRUE 或 FALSE 也可在会话中打开或者关闭 SQL 跟踪。

#### 3.1 事务处理与类型分析

TPC-C 工作负载中定义了 5 种事务类型：New-Order，Payment，Order-Status，Delivery 和 Stock-Level。其中至少 43% 是 Payment，Order-Status，Delivery 和 Stock-Level 分别至少是 4%，New-Order 最多是 45%。这个混合事务的大约 8% 是只读事务。New-Order 事务平均在 8 个表中进行  $2*k+2$  次选择、 $k+1$  次更新和  $k+2$  次插入，其中  $k$  是在新

订单中的行数，是一个在 5~ 15 之间平均分布的随机数（平均为 10）。Payment 事务小一些，平均在 4 个表中进行大约 5.3 次选择、3 次更新和 1 次插入。Order-Status 和 Stock-Level 事务是只读的，分别平均在 3 个表中进行 14 次选择和 3 个表中进行 401 次选择。Delivery 事务在 4 个表中进行 120 次选择、120 次更新和 10 次删除。如表 1 所示。

表 1 TPC-C 事务处理操作分析表

| 事务名称         | 最少百分比 | 涉及表数 | Selects | Updates | Inserts | Deletes |
|--------------|-------|------|---------|---------|---------|---------|
| New Order    | *     | 8    | $2*k+2$ | $k+1$   | $k+2$   | 0       |
| Payment      | 43    | 4    | 5.3     | 3       | 1       | 0       |
| Order Status | 4     | 3    | 14      | 0       | 0       | 0       |
| Delivery     | 4     | 4    | 120     | 120     | 0       | 10      |
| Stock Level  | 4     | 3    | 401     | 0       | 0       | 0       |

在分析事务操作的同时，得知，事物的这些操作大都集中在 Customer、Warehouse、District、New\_Order 等几个表上，因此这些表尤其是其中某一个表的 I/O 性能将影响到整个系统的性能，是优化工作的关键。

#### 3.2 数据库物理设计

事务项通过 LGWR 后台进程写入日志文件，事务数据同时写入一些表空间（如 RBS 和 DATA 表空间），表空间的写入是通过 DBWR 后台进程来实现的。如果数据文件同日志文件存在同一磁盘，则 DBWR 和 LGWR 后台进程的冲突是不可避免的。日志文件是连续写入，而数据文件的读写相对随机。控制文件的 I/O 很小，但是它们应当进行镜像，至少有 3 个拷贝，放在 3 个驱动器中。同样要注意 LGWR 与 ARCH 的冲突。数据文件与数据库代码间的冲突也是应当关注的另一问题。为减少这方面的 I/O 冲突，将系统数据库文件、用户数据库文件以及日志文件分别存放于不同的物理磁盘中。

在用户数据库内部，Oracle 对象（表和索引）设置也会影响磁盘 I/O，数据表 Pctfree、Pctused 值的设置权衡高效的数据块空间利用和操作性能之间的关系，设置原则是： $db\_block\_size * (100 - PCTFREE - PCTUSED)$  必须比行的长度大。Pctused 设置为  $PCTUSED = 100 - PCTFREE - \max(10, (\text{maximum-row-size} / \text{blocksize}) * 100)$ 。根据 TPC-C 测试系统各表行的长度、对这些表操作的不同种类以及麒麟操作系统中数据块的大小为 4k 等特点，在设计表时我们对如 New\_Order 表 Insert 后不久 Delete 的特点，采用小的 PCTFREE，非常小的 PCTUSED；对于如 District、Customer 等表以 Update 操作为主的特点，根据行长度的变化范围来确定 PCTFREE，PCTUSED 使用缺省值；而对于仅以 Insert 操作为主的表如 Order 表，则应设置小的 PCTFREE，缺省的 PCTUSED；以保证在保证效率的情况下，提高空间利用率。

#### 3.3 影响性能的 SQL 语句的调整措施

TPC-C 数据库包含的每张表，每个仓库所要求初始化的容量如表 2 所示。

表 2 TPC-C 测试表情况

| 表名称        | 属性数(个) | 容量(行) | 每行长度(B) | 表长度(kB) |
|------------|--------|-------|---------|---------|
| Warehouse  | 9      | 1     | 89      | 0.089   |
| District   | 11     | 10    | 95      | 0.950   |
| Customer   | 21     | 30k   | 655     | 19 650  |
| History    | 8      | 30k   | 46      | 1 380   |
| Order      | 8      | 30k   | 24      | 720     |
| New_Order  | 3      | 9k    | 8       | 72      |
| Order_Line | 10     | 300k  | 54      | 16 200  |
| Stock      | 17     | 100k  | 306     | 30 600  |
| Item       | 5      | 100k  | 82      | 8 200   |

从表 2 中不难看出，Customer 表结构复杂，有 21 个属性，容量大。当 TPC-C 进行 30 个仓库 (Warehouse) 测试时，该表容量达到 600M 和 100 万行。对该表的读写操作很复杂，合理建立该表索引能大幅地减少系统 I/O 量，提高整体的测试性能。

### 3.3.1 TPC-C 事务对 Customer 表操作分析

在 TPC-C 的 5 项事务中除占 4% 的 Stock\_Level 事务不对该表进行操作外，其余 4 个占总量 96% 的事务都要对 Customer 表不同属性进行读和写操作，是所有 9 个表中操作最频繁的表，其中对该表的读操作占绝大多数，而索引能大大地提高此类以查询为主要操作系统的效率。

### 3.3.2 TPC-C 事务中 Customer 表属性使用规则

根据 TPC-C 测试事务规则，对 Customer 表查询选择用户时，60% 的事务时间是根据用户姓氏 (C\_Last) 属性进行查询，40% 的事务时间根据用户号进行查询。按用户姓氏查询后，再根据 C\_First 进行排序查找最终结果。同时 C\_W\_ID、C\_D\_ID 也是每次查询的必要条件。在 Customer 表里，各属性作为查询条件使用频率依次为 C\_LAST, C\_W\_ID, C\_D\_ID, C\_ID, C\_FIRST。

综合以上对 Customer 的分析，TPC-C 测试标准中对 Customer 的原来的 Primary Key: (C\_W\_ID, C\_D\_ID, C\_ID), Foreign Key (C\_W\_ID, C\_D\_ID), references (D\_W\_ID, D\_ID) 进行调整，增加了一个索引具体操作为

```
CREATE UNIQUE INDEX C_CUST2
ON C_CUSTOMER ("C_LAST", "C_W_ID","C_D_ID","C_FIRST",
"C_ID")
TABLESPACE "USERS" PCTFREE 1 INITRANS 3
STORAGE ( BUFFER_POOL DEFAULT );
```

### 3.4 Oracle 数据库系统内存调整

以上主要针对的是 TPC-C 系统应用程序设计与 SQL 的调整，在分析调整了由于应用程序 SQL 等原因形成的系统性问题后，以下将从 Oracle 通常的如数据库内存结构、物理 I/O 等方面进行调整。

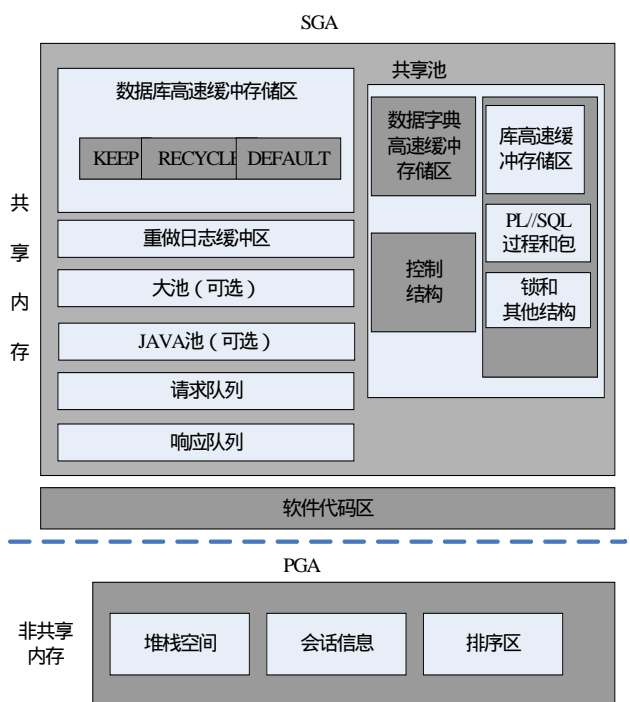


图 3 Oracle 内存结构

Oracle 服务器由一个实例 (Instance) 和一个数据库组

成。实例分配的内存结构主要由系统全局区 SGA 和程序全局区 PGA 构成，如图 3 所示。

在图 2 的测试环境中，使用单台联想服务器进行测试，Oracle 内存的缺省 SGA 的设置为：共享池为 48MB，缓冲区高速缓存为 144MB，大型池为 32MB 等，而请求队列和响应队列是用于多线程服务器配置。PGA 的缺省设置为 62MB。通过对这些内存参数进行分析并调整能有效地提高数据库系统对内存的利用率，从而达到提高系统性能的作用。

### 3.4.1 数据高速缓冲存储区 (Buffer Cache) 的调整

数据高速缓冲存储区存储了最近使用的从数据文件中读入的数据块的备份。例如当执行一个查询时，服务器进程从数据库高速缓存中查找所需要的块，假如没有找到就从数据文件中读入块，并将其复制到数据高速缓存中，后续的操作如果需要相同的块，则从内存中查找，而不必执行物理的读，能大大减少磁盘的 I/O，同时该部分的命中率决定了数据库的性能。在调整该块内存时为保障其最大命中率，减少磁盘 I/O，将除操作系统、应用程序、PGA 和少部分其他 SGA 内存块以外的绝大部分内存分配至该块。

### 3.4.2 共享池 (Share Pool) 的调整

共享池是在语法分析过程中使用的一部分内存，其中存储如最近执行的 SQL 命令、PL/SQL 过程和包、最近使用的数据字典数据、锁、字符集信息、安全特性等信息。因此，使用不同大小的共享池将影响语法分析占用的时间。在系统中经过多次 TPC-C 测试后，发现当共享池内存设置为 20MB 直至 128MB 实现同样的节省时间 163s 和节省时间因子 1 的效果。同时使用 Select SUM(pins) ,SUM(reloads) from V\$librarycache; 当共享池调整为 64MB 时，SUM(reloads)/SUM(pins) 0，说明该值较为合适。

### 3.4.3 大型池 (Large Pool) 的调整

SGA 中大型池是一个可选区域，是为了实现备份与恢复这种特定的数据库操作而提供的较大的内存分配，大池允许 Oracle 从单独的缓冲池中获得较大的内存分配，而避免多个应用对相同内存的竞争。在测试系统中基本上不考虑数据库的备份和恢复，将该部分内存调整为 16MB。

### 3.5 I/O 监控与调整

在进行 TPC-C 测试时，通过 Statepack 对系统进行监控，从报表的首要 5 个等待事件列表中的 Db file sequential read 和 Enqueue 两个等待事件占了总等待时间的 96.68%，而 Db file sequential read 通常是指单一的数据块读操作 (例如索引的读取)。该值过大说明表的连接顺序很糟糕，或者使用了非选择性索引。进一步从 Stats\$filestatxs 表中获取的统计数据可得到文件层次上的读入和写入的活动细节。获取这些信息后，可以得知测试系统主要 I/O 瓶颈出现在如 Customer 表索引的创建不合理等原因上。除了进行相应的索引调整外，在 Oracle 实例中，将 da\_block\_size 设置值加大，这因为块尺寸越大，磁盘 I/O 就越少。将 db\_block\_buffers 数据缓冲的数量加大，使得 Oracle 需要进行磁盘 I/O 的机会就越小，使得测试系统 I/O 更加顺畅，提高系统性能。

### 3.6 性能调优后的结果

通过以上调整在原有测试环境中 TPC-C 程序稳定地实现了 31 个 Warehouse 的测试运行，TPMC 值为 354.7，TPMC 值提高了 55.26%。

(下转第 87 页)