

Oracle Performance Tuning

Oracle 数据库性能优化

盖国强 冯春培 叶梁 冯大辉 编著

- ☑ 如何学习Oracle
- ☑ DBA优化之路
- ☑ Statspack高级调整
- ☑ 使用SQL_TRACE/10046事件进行数据库诊断
- ☑ 表空间的存储管理与优化技术
- ☑ 关于Freelists和Freelist Groups的研究
- ☑ 深度分析数据库的热点块问题
- ☑ Shared Pool原理及性能分析
- ☑ 捕获问题SQL 解决过多CPU消耗问题
- ☑ Web分页与优化技术



人民邮电出版社
POSTS & TELECOM PRESS

TP311.1380
G011

Oracle Performance Tuning

Oracle

数据库性能优化

盖国强 冯春培 叶梁 冯大辉 编著

人民邮电出版社

图书在版编目(CIP)数据

Oracle 数据库性能优化 / 盖国强等编著. —北京: 人民邮电出版社, 2005.6

ISBN 7-115-13438-3

I. O... II. 盖... III. 关系数据库—数据库管理系统, Oracle IV. TP311.138

中国版本图书馆 CIP 数据核字 (2005) 第 048495 号

内 容 提 要

本书面向实际应用, 从多个角度出发, 对 Oracle 优化中的很多关键问题进行了深入全面的探讨, 涵盖了 Oracle 优化的各个技术层面, 从内存优化、IO 规划及优化, 到 SQL 优化调整, 以较为完整的体系阐述了 Oracle 的优化技术。

本书给出了大量取自实际工作现场的实例。在分析实例的过程中, 兼顾深度与广度, 不仅对实际问题的现象、产生原因和相关的原理进行了深入浅出的讲解, 更主要的是, 结合实际应用环境, 提供了一系列解决问题的思路和方法, 包括详细的操作步骤, 具有很强的实战性和可操作性, 满足面向实际应用的读者需求。

Oracle 数据库性能优化

-
- ◆ 编 著 盖国强 冯春培 叶 梁 冯大辉
责任编辑 杜 洁
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京密云春雷印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787×1092 1/16
印张: 31.5
字数: 763 千字 2005 年 6 月第 1 版
印数: 4 001—6 000 册 2005 年 8 月北京第 2 次印刷

ISBN 7-115-13438-3/TP · 4674

定价: 65.00 元

读者服务热线: (010)67132692 印装质量热线: (010)67129223



前言

关于本书

全球著名的 IT 及电信行业市场咨询和顾问机构 IDC 公司的统计数据(2005 年 3 月)表明,2004 年 Oracle 数据库市场年增长率为 14.5%,且以 41.3%的市场占有率居市场领先地位。与此同时,Oracle 数据库在国内也得到了进一步的发展和普及。

应用的普及势必对数据库的管理及优化提出了更高的要求,而 Oracle 数据库的性能优化一直是数据库管理中的重要环节,也是最复杂的内容之一。基于此,我们组织编写了本书。

作为国内著名 Oracle 技术论坛,ITPUB (<http://www.itpub.net>)一直致力于推动各种形式的技术互动、促进经验交流和提供多种共享资源,以期帮助更多的朋友学习并掌握 Oracle 技术。2004 年,ITPUB 首次编写自己的技术图书《Oracle 数据库 DBA 专题技术精粹》,得到了广大读者的支持与鼓励。

本书是 ITPUB 推出的第二本技术图书,由活跃在 ITPUB 上的一群资深 Oracle 技术人员组稿,历时一年多的努力而成,其主要内容都是来自于实践经验的提炼和总结,具有高度的实用性及参考性。

本书特点

本书内容来自于实践,又高于实践,不但分析实际工作中与性能相关的问题及原理,还提供了详细的操作步骤,兼顾深度与广度,具有很强的实战性和可操作性。

本书对 Oracle 优化中的很多关键问题进行了深入的探讨,其探索程度及实用性在目前市场上的同类书籍中还是比较罕见的。不管是一个初学者还是想深入学习 Oracle 优化技术的专业人士,都能在这本书中找到自己感兴趣的部分。

本书作者

本书集中了 ITPUB 的技术优势,是集体创作的结晶,由 ITPUB 四大技术版主盖国强、冯春培、叶梁和冯大辉担当主编, Richard J Niemiec、吕学勇、陈吉平、段凌、何小栋、李轶楠、汪海、谢中辉、邢海捷、杨廷琨、

叶宇、张乐奕、陈宇红、诸超、张大鹏、张春宏、张权（排名不分先后）等参与了本书的编写。他们都是来自各行各业的资深数据库管理人员，具有丰富的实践经验。同时感谢他们为此书所付出的努力和心血。

值得一提的是，在本书即将完稿时，我们收到了 Richard J. Niemiec 先生委托吕学勇先生带来的文章“Statspack 高级调整”，现在把这篇文章收入本书，以便更多的读者可以从中受益；我们也欣喜地看到 DBA 渐无国界，希望国内的数据库从业者能够渐渐走向国际。

本书结构

本书分为 5 篇，共 30 章，具体结构划分如下。

☞ 第一篇：优化工具篇，包括第 1~6 章。这部分内容对 Oracle 数据库优化中常用的工具进行介绍，并辅以案例说明。

☞ 第二篇：存储优化篇，包括第 7~12 章。这部分内容主要对于存储及 IO 优化相关问题进行探讨。

☞ 第三篇：内存调整篇，包括第 13~17 章。这部分内容对于内存结构、原理以及优化调整进行了分析。

☞ 第四篇：诊断案例篇，包括第 18~24 章。这部分包含了实践中经常遇到的问题及案例诊断，通过具体案例介绍解决问题的方法及思路。

☞ 第五篇：SQL 优化及其他篇，包括第 25~32 章。这部分内容对于实践中棘手的 SQL 优化问题及相关问题进行了探讨。

本书的读者对象

本书适用于具有一定数据库基础的数据库从业人员，尤其适用于 Oracle 数据库管理和开发人员，也可以作为各大中专院校相关专业的教学辅导和参考用书，或作为相关培训机构的培训教材。

本书约定

1. 为了简便，本书在不影响读者阅读的前提下采用了简称，例如，“Oracle 9i”简称“9i”，“Oracle 8i”简称“8i”等。

2. 为了给读者提供更多的学习资源，同时弥补本书篇幅有限的遗憾，本书提供了大量的参考链接，许多本书无法详细介绍的问题都可以通过这些链接找到答案。因为这些链接地址会因时间而有所变动或调整，所以在此说明，这些链接信息仅供参考，本书无法保证所有的这些信息是长期有效的。

3. 本书所列出的插图、运行结果可能会与读者实际环境中的操作界面有所差别，这可能是由于操作系统平台、Oracle 版本的不同而引起的，在此特别说明，一切以实际情况为准。

4. 广大读者如有好的建议或在学习本书中遇到疑难问题，欢迎到网站 (<http://www.itpub.net>) 进行探讨，也可发电子邮件联系我们 (magazine@itpub.net)。

编者
2005.4



Oracle 学习之路

如何学习 Oracle 一直是一个被普遍关注的话题，在网上我们也能看到很多关于这方面的讨论，这本身就是一个巨大的进步。如果回到三四年前，你会发现学习 Oracle 通常是个人行为，很少的学习资料，很少的讨论，很多时候你需要一个人面对各种复杂的环境。

可是，网络迅速地改变了这一切。现在你可以很容易地在网络上找到你需要的各类资料，也可以很容易地加入各种讨论。ITPUB (<http://www.itpub.net>) 就是这样一个能够提供各种资源的场所，也因此才有了本书的出版。

本书的两位主编在关于如何学习 Oracle 方面发表了自己的看法，希望对大家有所帮助有所借鉴。

如何学习 Oracle——盖国强 (eygle) 如是说

经常有朋友会问，应该如何学习 Oracle，怎样才能快速提高？我把自己的一点心得写在这里，供大家参考。

其实学习任何东西都是一样，没有太多的捷径可走，必须打好了坚实的基础，才有可以在进一步学习中得到快速提高。

王国维在他的《人间词话》中曾经概括了为学的三种境界，我在这里套用一下：

古今之成大事业、大学问者，罔不经过三种之境界。

“昨夜西风凋碧树。独上高楼，望尽天涯路。”此第一境界也。

“衣带渐宽终不悔，为伊消得人憔悴。”此第二境界也。

“众里寻他千百度，蓦然回首，那人却在灯火阑珊处。”此第三境界也。

这也是你学习 Oracle 必须经历的三种境界。

第一层境界是说，学习的路是漫漫的，你必须做好充分的思想准备，如果半途而废还不如不要开始。

这里，注意一个“尽”字，在开始学习的过程中，必须充分阅读 Oracle 的基础文档、概念手册、管理手册、备份恢复手册等（这些文档都可以在 <http://tahiti.oracle.com> 上找到）；OCP 认证的教材也值得仔细阅读。打好基础之后，你才具备了进一步提升的能力，万丈高楼都是由地而起的。

第二层境界是说，尽管经历挫折、打击、灰心、沮丧，也都要坚持不放弃。具备了基础知识之后，你可以对自己感兴趣或者工作中遇到的问题进行深入思考，由浅入深从来都不是轻而易举的，甚至很多时候会感到自己停滞不前了，但是不要动摇，学习及理解上的突破是会需要一些时间的。

第三次境界是说，经历了那么多努力以后，你会发现，那苦苦思考的问题，那百思不得其解的算法原理，原来答案就在眼前，你的思路豁然开朗，宛如拨云见日。这个时候，学习对你来说，不再是个难题，也许是种享受，也许成为一种艺术。

所以如果你想问我如何速成，那我是没有答案的。不经一番寒彻骨，哪得梅花扑鼻香。当然这三种境界在实际中也许是交叉进行的，在不断的学习中，不断有蓦然回首的惊喜。

我自己在学习的过程中，经常是采用“由点及面”的方法。

当遇到一个问题后，一定要深入下去，穷究根本，这样你会发现，一个简单的问题也会带起一大片的知识点，如果能对很多问题进行深入思考和研究，那么在深处，你会发现，这些问题逐渐接合，慢慢地延伸到 Oracle 技术的所有层面，逐渐地就能融会贯通。这时候，你会主动地去尝试全面学习 Oracle，扫除自己的知识盲点，学习已经成为一种需要。

由实践触发的学习才最有针对性，才能让自己更深入地理解书本上的知识，正所谓“纸上得来终觉浅，绝知此事要躬行”。实践的经验才是我们最为宝贵的。

如果说有捷径，那么这就是我的捷径。想想自己，经常是“每有所获，便欣然忘食”，兴趣才是我们最好的老师。

Oracle 的优化是一门学问，也是一门艺术，理解透彻了就会知道，优化不过是在各种条件之下所做出的均衡与折中。

内存、外存、CPU、IO……对这一切你都需要有充分的认识和相当的了解，管理数据库所需要的知识并不单纯。作为一个数据库管理人员，需要做的就是能够根据自己的知识以及经验在各种复杂情况下做出快速正确的判断。当问题出现时，需要知道使用怎样的手段发现问题的根本；在找到问题之后，需要运用你的知识找到解决问题的方法。这当然并不容易，举重若轻还是举轻若重，取决于自己具备怎样的基础以及经验积累。

在网络上, Howard J. Rogers 最近创造了一个新词组: Voodoo Tuning, 用以形容那些没有及时更新自己知识技能的所谓的 Oracle 技术专家。由于知识的陈旧或者理解的肤浅, 这些专家提供的很多调整建议可能是错误的、容易使人产生误解。如果愿意回到 Oracle 5 版或者 6 版的年代, 他们提供的某些建议在有些情况下也许是正确的。但是这些建议在 Oracle 7.0/8.0 或者 Oracle 8i 以后的版本中往往是完全错误的。

后来基于类似问题触发了互联网内 Oracle 顶级高手的一系列深入讨论, TOM、Jonathan Lewis、HJR 等人都参与其中, 在我的网站 (<http://www.eygle.com>) 上对这些内容及相关链接作了简要的介绍, 有兴趣的读者可以参考。

HJR 给我们提出了一个很好的提示: 对于需要调整的内容, 你必须具有充分的认识, 否则做出的判断就有可能是错误的。这也是我想给大家的一个建议: 学习和研究 Oracle, 严谨和认真必不可少。

当然你还需要勤奋, 我所熟悉的在 Oracle 领域有所成就的技术人员, 他们共同的特点就是勤奋。如果你觉得掌握的东西没有别人多, 那么也许就是因为你不如别人勤奋。

要是你觉得这一切过于复杂了, 那我还有一句简单的话送给大家: 不积跬步, 无以至千里。学习正是在逐渐积累过程中的提高。

现在 ITPUB 给我们提供了很好的交流场所, 很多问题都可以在这里找到答案, 大家可以在此互相讨论, 互相学习, 共同进步。这是我们的幸运, 同时也要感谢这个网络时代为我们营造了良好的学习环境和机会。

ITPUB 的第二本书即将出版, 谨以此祝愿 ITPUB 越来越好, 也愿我们的书能给大家带来知识和帮助。

关于学习 Oracle——冯春培 (biti_rainy) 如是说

我自己学习 Oracle 的经历, 无甚诀窍, 兴趣是首要, 持之以恒是关键。我几乎仔细阅读过 ITPUB 中绝大部分与数据库管理相关的帖子, 精华区的帖子更是没有放过, 或者仔细阅读一些有特点 ID 参与过的帖子, 多和人交流讨论, 明白的道理要讲给别人听并且让他们也同样明白, 这样自己才能记忆深刻, 多想, 多做实验, 当感觉不到提高的时候要继续坚持, 有时可能需要半年才会感觉到提高……这么反复几回就好了。当然这个过程说起来容易做起来难, 几年来坚持学习、思考、实验, Oracle 占用的时间几乎比睡觉的时间还多。

学习的过程是一个从点到线, 从线到网, 从网到面的过程。我自己的感觉非常明显, 先是枯燥的独立的知识点, 逐渐地连成线, 穿起来, 这是第一次感到有很大的进步, 再逐渐地在知识中建立了横向联系, 这样就形成了网状结构, 这个时候就有豁然开朗的感觉, 原来很

多以前看起来不相干的东西之间居然都有各种各样的联系。但这个时候发现，在 Oracle 知识网中还有细微的知识漏洞，等把这些漏洞弥补上后就成了面。这个时候觉得你自己是在玩 Oracle。在整个学习的过程中，我个人大约经历了 4 次明显的层次跃迁，每次跃迁之间大抵至少有半年的时间间隔。然后逐渐了解了一些 OS 的知识，发现和 Oracle 有异曲同工之处。我们来看 OS 和 Oracle 设计之间的相通之处。

OS 和 Oracle 都一样，在空间和时间的资源消耗中寻找平衡。试图去理解设计者的意图，是很重要的学习方法。比如 Oracle 10g 的 Flashback Database，为什么推出这个功能，是为了节约恢复时间，那么它的代价是什么，就是增加了磁盘空间消耗来记录变化的 block，这样就利用增加 IO 和存储的代价（包括数据库运行消耗的 CPU 时间）来节约了恢复的时间和资源。随着硬件性能和容量的不断提升，思想已经转变，把正常运行中可以拿出来的资源（对于 10 年前的 Oracle 来说内存、磁盘、CPU 都是需要尽量节约的资源）来弥补恢复时间太长的不足。

对比 Oracle 和 OS 的几个方面。

(1) Oracle 的 ASSM 在逻辑上和文件系统相似，当然和 Linux 下 ext3 思想更相近一些。

OS 本来是根据 512 byte block 而作为扩展单元来防止碎片（节约磁盘）的，但现在存储容量的发展早已日新月异，不再需要为几 KB 的空间费劲了，ext2/ext3 就是类似 Oracle 的 extent 来作为扩展单元，这样会更有利于 IO。这方面朝数据库走近了一步。而 Oracle 由 DMT 走向 LMT 再走向 ASSM，如果把 extent 看作 block 的话，有点类似索引的结构，或者说类似文件系统的 i-node 索引结构。这个可以看作数据库和 OS 都朝中间走了一步。

(2) Oracle 的 Buffer Cache 的管理和 OS 共享内存的管理思想很相似。

这个我更倾向于说 LRU/hash table/list。对于 SUN Solaris 来说，use_ism 本质上就是允许多个进程共享页表入口访问 page list，以达到并发的目的，跟 Oracle 的 Cache Buffer Chain 类似了。

(3) Oracle slaves 就是模拟 AIO。

Oracle 假如不使用 AIO，则 DBWR 是 write /wait /write /wait，下一个写必须等上一个写完成，而 io_slaves 的作用则是 DBWR 把要写的 buffer 交给 slaves，由 slaves 完成写，那么写还没有完成的时候 DBWR 可以去做下一步事情，搜集 dirty buffer 并提交下一个写请求。而 Oracle 的 AIO 概念，就是让 DBWR 可以 write/write/write，不必等上一个写完成就可以提交下一个写请求，达到并行写的目的，以充分利用空闲的硬件资源。

Oracle 跟 OS 还有其他很多方面的相似点，Oracle 的 redo log 和文件系统的 log 作用类似，Oracle 的事务和 OS 文件系统的事务类似，

Oracle DMT 的数据字典表、LMT 的 Bitmap/Extent Map 和文件系统的 i-node 类似。数据库这些空间控制的变化和日志文件系统的 Intent Log，在内存管理方面都使用 Hash table + List。

每个人在学习的过程中，总会总结出适合自己的一些方式、方法。要逐渐领悟数据库设计的原理，理解为什么要这么设计，这是为了解决一个什么样的问题，而不仅仅是表面上会使用数据库就止步。如此坚持下去，就会体会到学习的无穷乐趣，进步也会很快。

给大家的话

大家从以上两位主编的话中应该可以看到很多共同的东西：

兴趣 + 勤奋 + 坚持 + 方法 $\approx$ 成功

很遗憾，不能给这个公式画上“=”，但是无关紧要，只要具备了以上因素，每个人都会离成功不远。

我们祝愿大家离成功越来越近。



目 录

第 1 章	DBA 优化之路	3
1.1	学习的建议	3
1.2	工具推荐	4
1.3	关于操作系统方面的建议	4
1.4	关于 Oracle 初始化参数的调整	5
1.5	关于 Statspack 的若干建议	6
1.6	关于 logmnr 在调优中的运用	7
1.7	关于 materialized view 在调优中的运用	8
1.8	关于 Stored Outline 在 SQL 优化中的运用	8
1.9	用 dbms_profiler 调优存储过程	8
1.10	优化前的准备工作	9
1.11	如何对 SQL 进行调整及优化	10
1.12	表结构优化实例	12
1.13	如何对 session 进行跟踪	12
1.14	基于等待事件的性能诊断方法	13
1.15	基于资源限制的性能诊断方法	14
1.16	如何减少共享池的碎片	15
1.17	监控表及索引的意义	17
1.18	通过优化 SQL 消除 temp 表空间膨胀	17
1.19	理解 compress 选项在优化上的作用	19
1.20	关于在线重定义 table 的建议	19
1.21	关于分区表在数据库设计时的建议	20

1.22 关于 DataGuard 在高可用方面的建议	20
第 2 章 Statspack 高级调整	23
2.1 Statspack 高级调整译文	24
2.1.1 Top 5 Wait Events	24
2.1.2 等待时间快捷参考	26
2.2 关于 Latch	27
第 3 章 Statspack 使用的几个误区	33
3.1 以命中率为主衡量性能问题	33
3.2 快照的采样时间间隔问题	35
3.3 以偏概全	36
3.4 关于 TIMED_STATISTICS 参数的设定	36
3.5 你成了泄密者	37
第 4 章 TKPROF 工具使用简介	39
4.1 TKPROF 工具简介	39
4.2 TKPROF 工具的使用步骤	40
4.3 TKPROF 工具如何分析 trace 文件	41
第 5 章 使用 Oracle 的等待事件检测性能瓶颈	45
5.1 判断等待事件的相关视图	46
5.1.1 系统级统计信息 v\$system_event	46
5.1.2 会话级统计信息 v\$session_event	48
5.1.3 会话详细性能信息 v\$session_wait	48
5.1.4 会话等待事件的相关视图之间的关系	50
5.2 应该怎么考虑进行优化	50
5.3 主要等待事件	51
5.4 案例分析	54
5.5 小结	59
5.6 附录	59
第 6 章 使用 SQL_TRACE/10046 事件进行数据库诊断	63
6.1 SQL_TRACE 及 10046 事件的基础介绍	63
6.1.1 SQL_TRACE 说明	63
6.1.2 10046 事件说明	67
6.1.3 获取跟踪文件	68
6.1.4 读取当前 session 设置的参数	68
6.2 案例分析之一	69
6.2.1 问题描述	69

6.2.2	检查并跟踪数据库进程	69
6.2.3	检查 trace 文件	70
6.2.4	登录数据库检查相应表结构	71
6.2.5	解决方法	72
6.2.6	小结	73
6.3	案例分析之二	73
6.3.1	问题描述	73
6.3.2	drop user 出现问题	74
6.3.3	跟踪问题	74
6.3.4	问题定位	76
6.3.5	实际处理	77
6.3.6	小结	78
6.4	10046 与等待事件	78
6.4.1	10046 事件的使用	78
6.4.2	10046 与 db_file_multiblock_read_count	80
6.4.3	10046 与执行计划的选择	82
6.4.4	db_file_multiblock_read_count 与系统的 IO 能力	83
6.4.5	小结	85

第二篇 存储优化篇

第 7 章	表空间的存储管理与优化技术	89
7.1	表空间的作用与分类	89
7.2	字典管理表空间	90
7.2.1	字典管理表空间的特性	90
7.2.2	字典管理表空间的缺点	91
7.2.3	字典管理表空间的优化	92
7.3	本地管理表空间	92
7.3.1	本地管理表空间的特性	92
7.3.2	管理位图块的内部结构	94
7.3.3	本地管理表空间的优点	94
7.4	段自动管理表空间	95
7.4.1	段自动管理表空间的特性	95
7.4.2	位图管理段内部结构	96
7.4.3	段自动管理表空间的优化	98
7.5	9i 对表空间的管理优化	98
7.5.1	自动 undo 管理的表空间	98
7.5.2	完全本地的临时表空间	99
7.6	Oracle 10g 对表空间的优化	99

7.7 小结	99
7.8 附录	99
第 8 章 关于 Oracle 数据库中行迁移/行链接的问题	101
8.1 行迁移/行链接的简介	101
8.2 行迁移/行链接的检测方法	106
8.3 行迁移/行链接的清除方法	108
第 9 章 HWM 与数据库性能的探讨	121
9.1 什么是 HWM	121
9.2 初始创建的 table 中 HWM 的不同情况	122
9.3 insert 数据时 HWM 的移动	128
9.4 HWM 对性能的影响	131
9.5 何时应该降低 HWM	135
9.5.1 对于 LMT 下的 FLM	135
9.5.2 对于 ASSM	136
9.6 如何降低 HWM	137
9.6.1 Move	137
9.6.2 DBMS_REDEFINITION	142
9.6.3 Shrink	143
9.6.4 小结	148
9.7 其他几种会移动 HWM 的操作	148
9.7.1 Insert Append	148
9.7.2 Truncate	152
第 10 章 调整 I/O 相关的等待	153
10.1 Oracle 数据库 I/O 相关竞争等待简介	153
10.2 Oracle 数据库 I/O 相关竞争等待的处理方法	154
10.3 Oracle 数据库 I/O 相关的等待事件和相应的解决方法	157
10.3.1 数据文件相关的 I/O 等待事件	158
10.3.2 控制文件相关 I/O 等待事件	163
10.3.3 重做日志文件相关的等待事件	164
10.3.4 高速缓存区相关的 I/O 等待事件	166
10.4 小结	169
第 11 章 Oracle 在 Solaris 的 VxFS 上的异步 I/O 问题	171
11.1 VxFS 文件系统的简介	171
11.2 VxFS 文件系统上如何启用异步 I/O	171
11.3 如何检测在 VxFS 文件系统上是否支持异步 I/O	172
11.4 如何查看 VxFS 文件系统上异步 I/O 的性能	173

11.5 如何转换 VxFS 文件系统上数据文件为支持异步 I/O 的数据文件	174
第 12 章 关于 Freelists 和 Freelist Groups 的研究	177
12.1 什么是 Freelists	177
12.2 Freelists 是否已经过时	178
12.3 Freelists 存储在哪里	178
12.4 有多少种 free list	180
12.5 进程请求空闲块的过程	182
12.6 块在 free list 间的移动	184
12.7 关于 free list 将导致大量空间浪费的误解	185
12.8 关于 Freelists 和 Freelist Groups 的一个比喻	186
12.9 与 Freelists 和 Freelist Groups 相关的等待事件	186
第三篇 内存调整篇	
第 13 章 自动 PGA 管理——原理及优化	193
13.1 什么是 PGA 内存自动管理	193
13.2 PGA Advice 功能	199
13.3 自动 PGA 内存管理相关初始化参数	201
第 14 章 32bit Oracle SGA 扩展原理和 SGA 与 PGA 的制约关系	203
14.1 如何识别 32bit 的 Oracle	203
14.2 为何存在 1.7GB 的限制	204
14.3 32bit 下 SGA 与 PGA 之间的制约关系	207
第 15 章 KEEP 池和 RECYCLE 池	213
15.1 Oracle 的数据缓冲池	213
15.2 KEEP 池和 RECYCLE 池	214
15.2.1 KEEP 池	215
15.2.2 RECYCLE 池	219
15.3 小结	221
第 16 章 深度分析数据库的热点块问题	223
16.1 热点块的定义	223
16.2 数据缓冲区的结构	223
16.3 如何确定热点对象	224
16.4 热点问题的解决	228
16.5 热点块的其他相关症状	230

166 小结	231
第 17 章 Shared Pool 原理及性能分析	233
17.1 Shared Pool 的基本原理	233
17.2 Shared Pool 的设置说明	233
17.2.1 基本知识	234
17.2.2 Shared Pool 的 Free List 管理	235
17.2.3 了解 X\$KSMSPP 视图	240
17.3 诊断和解决 ORA-04031 错误	244
17.3.1 什么是 ORA-04031 错误	244
17.3.2 内存泄露	245
17.3.3 绑定变量和 cursor_sharing	246
17.3.4 使用 Flush Shared Pool 缓解共享池问题	247
17.3.5 shared_pool_reserved_size 参数的设置及作用	247
17.3.6 其他	249
17.3.7 模拟 ORA-04031 错误	249
17.4 Library Cache Pin 及 Library Cache Lock 分析	252
17.4.1 Library Cache Pin 等待事件	253
17.4.2 Library Cache Lock 等待事件	258
17.5 诊断案例一	259
17.6 诊断案例二	267
17.7 小结	269

第四篇 诊断案例篇

第 18 章 一次性能调整过程总结	273
18.1 系统环境	273
18.2 基本的调优过程	273
18.2.1 db file scattered read	273
18.2.2 db file sequential read	274
18.2.3 Enqueue	275
18.2.4 Latch Free	275
18.3 小结	281
第 19 章 电信业 Oracle 优化手记	283
19.1 一条 SQL 语句要运行 2 年怎么办	283
19.2 优化的传统定律和新时尚	285
19.2.1 index 和表同一个表空间（过时）	286
19.2.2 定期重建索引（过时）	287

19.2.3 裸设备应该取代文件系统（过时）	287
19.2.4 初始参数设置 cursor_sharing=similar（不一定有效）	288
19.2.5 初始参数设置 fast=true（有效）	289
19.3 联机重做日志的优化	289
19.3.1 联机重做日志组内创建多个成员	289
19.3.2 加大 redo log 的容量	290
第 20 章 一次诊断和解决 CPU 利用率高的问题分析	291
20.1 问题的具体描述	291
20.2 问题的详细诊断解决过程	292
20.3 小结	296
第 21 章 一次异常内存消耗问题的诊断及解决	297
21.1 问题发现	297
21.2 解决过程	297
21.2.1 环境介绍	297
21.2.2 问题现象	297
21.2.3 对比分析	299
21.2.4 假设和分析	300
21.2.5 找到根源	302
21.2.6 解决问题	303
21.3 小结	305
第 22 章 如何捕获问题 SQL 解决过度 CPU 消耗问题	307
22.1 检查当前情况	307
22.2 使用 Top 工具辅助诊断	308
22.3 检查进程数量	309
22.4 登录数据库	309
22.5 捕获相关 SQL	311
22.6 创建新的索引以消除全表扫描	313
22.7 观察系统状况	314
22.8 性能何以提高	315
22.9 小结	317
第 23 章 一条 SQL 导致数据库整体性能下降的诊断及解决	319
23.1 现象	319
23.2 诊断与解决	319