

<<Oracle SQL高级编程>>

图书基本信息

书名：<<Oracle SQL高级编程>>

13位ISBN编号：9787115266149

10位ISBN编号：711526614X

出版时间：2011-11

出版时间：人民邮电出版社

作者：[美] Karen Morton, Kerry Osborne, Robyn Sands, Riyaj Shamsudeen, Jared Still

页数：502

译者：朱浩波

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Oracle SQL高级编程>>

内容概要

Oracle

数据库中的SQL是当今市场上功能最强大的SQL实现之一，而本书全面展示了这一工具的威力。如何才能让更多人有效地学习和掌握SQL呢？

Karen

Morton及其团队在本书中提供了专业的方案：先掌握语言特性，再学习Oracle为提升语言效率而加入的支持特性，进而将两者综合考虑并在工作中加以应用。

作者通过总结各自多年的软件开发和教学培训经验，与大家分享了掌握Oracle

SQL所独有的丰富功能的技巧所在，内容涵盖SQL执行、联结、集合、分析函数、子句、事务处理等多个方面。

读者可以学习到以下几个方面的技巧：

- 掌握Oracle数据库中独有的SQL强大特征；
- 读取并理解SQL执行计划；
- 快速分析并改进表现欠佳的SQL；
- 通过提示及配置文件等来控制执行计划；
- 在程序中优化查询而无需改动代码。

作为Oracle

SQL经典著作之一，本书为SQL开发人员指明了前行的方向，赋予了他们不断开拓的动力。

<<Oracle SQL高级编程>>

作者简介

KAREN MORTON

研究人员、教育家及顾问，Fidelity信息服务公司的资深数据库管理员和性能调优专家。

她从20世纪90年代初就开始使用Oracle，从事

Oracle的教学工作也已经超过10年的时间。

她是Oracle

ACE，也是OakTable（Oracle社区中著名的“Oracle科学家”的非正式组织）的成员，经常在技术会议上演讲。

她的著作还包括

Expert Oracle Practices和Beginning Oracle

SQL，博客主页是karenmorton.blogspot.com。

KERRY OSBORNE

专注于Oracle咨询的Enkitec公司的创始人之一。

从1982年开始使用Oracle（第2版）。

他当过开发人员，也做过DBA，目前是

Oracle ACE总监和OakTable成员。

最近几年，他专注于研究Oracle内部原理以及解决性能问题。

他的博客主页是

kerryosborne.oracle-guy.com。

ROBYN SANDS

思科公司的软件工程师，为思科的客户设计开发嵌入式Oracle数据库产品。

从1996年开始使用Oracle，在应用开发、大型系统实现以及性能评估方面具有丰富的经验。

她是OakTable的成员，同时是Expert

Oracle Practices（2010年Apress出版）一书的合著者。

RIYAJ SHAMSUDEEN

专注于性能/数据恢复/电子商务的咨询公司OralInternals的首席数据库管理员和董事长。

有近20年使用Oracle技术产品以及Oracle数据库管理员/Oracle数据库应用管理员的经验，是真正应用集群、性能调优以及数据库内部属性方面的专家。

同时是一位演讲家及Oracle

ACE。

JARED STILL

从1994年就开始使用Oracle。

他认为对于SQL的学习是永无止境的，相信每一个查询Oracle数据库的人都需要精通SQL语言，才能写出高效的查询。

他参与本书的编写就是为了帮助别人实现这一目标。

<<Oracle SQL高级编程>>

书籍目录

第1章 SQL 核心

1.1 SQL 语言

1.2 数据库的接口

1.3 SQL*Plus 回顾

1.3.1 连接到数据库

1.3.2 配置SQL*Plus 环境

1.3.3 执行命令

1.4 5 个核心的SQL 语句

1.5 SELECT 语句

1.5.1 FROM 子句

1.5.2 WHERE 子句

1.5.3 GROUP BY 子句

1.5.4 HAVING 子句

1.5.5 SELECT 列表

1.5.6 ORDER BY 子句

1.6 INSERT 语句

1.6.1 单表插入

1.6.2 多表插入

1.7 UPDATE 语句

1.8 DELETE 语句

1.9 MERGE 语句

1.10 小结

第2章 SQL 执行

2.1 Oracle 架构基础

2.2 SGA-共享池

2.3 库高速缓存

2.4 完全相同的语句

2.5 SGA-缓冲区缓存

2.6 查询转换

2.7 视图合并

2.8 子查询解嵌套

2.9 谓语句前推

2.10 使用物化视图进行查询重写

2.11 确定执行计划

2.12 执行计划并取得数据行

2.13 SQL 执行——总览

2.14 小结

第3章 访问和联结方法

3.1 全扫描访问方法

3.1.1 如何选择全扫描操作

3.1.2 全扫描与舍弃

3.1.3 全扫描与多块读取

3.1.4 全扫描与高水位线

3.2 索引扫描访问方法

3.2.1 索引结构

<<Oracle SQL高级编程>>

- 3.2.2 索引扫描类型
- 3.2.3 索引唯一扫描
- 3.2.4 索引范围扫描
- 3.2.5 索引全扫描
- 3.2.6 索引跳跃扫描
- 3.2.7 索引快速全扫描
- 3.3 联结方法
 - 3.3.1 嵌套循环联结
 - 3.3.2 排序—合并联结
 - 3.3.3 散列联结
 - 3.3.4 笛卡儿联结
 - 3.3.5 外联结
- 3.4 小结
- 第4章 SQL 是关于集合的
 - 4.1 以面向集合的思维方式来思考
 - 4.1.1 从面向过程转变为基于集合的思维方式
 - 4.1.2 面向过程vs.基于集合的思维方式：一个例子
 - 4.2 集合运算
 - 4.2.1 UNION 和UNION ALL
 - 4.2.2 MINUS
 - 4.2.3 INTERSECT
 - 4.3 集合与空值
 - 4.3.1 空值与非直观结果
 - 4.3.2 集合运算中的空值行为
 - 4.3.3 空值与GROUP BY 和ORDER BY
 - 4.3.4 空值与聚合函数
 - 4.4 小结
- 第5章 关于问题
 - 5.1 问出好的问题
 - 5.2 提问的目的
 - 5.3 问题的种类
 - 5.4 关于问题的问题
 - 5.5 关于数据的问题
 - 5.6 建立逻辑表达式
 - 5.7 小结
- 第6章 SQL 执行计划
 - 6.1 解释计划
 - 6.1.1 使用解释计划
 - 6.1.2 理解解释计划可能达不到的目的的方式
 - 6.1.3 阅读计划
 - 6.2 执行计划
 - 6.2.1 查看最近生成的SQL 语句
 - 6.2.2 查看相关执行计划
 - 6.2.3 收集执行计划统计信息
 - 6.2.4 标识SQL 语句以便以后取回计划
 - 6.2.5 深入理解DBMS_XPLAN 的细节
 - 6.2.6 使用计划信息来解决问题

<<Oracle SQL高级编程>>

6.3 小结

第7章 高级分组

7.1 基本的GROUP BY 用法

7.2 HAVING 子句

7.3 GROUP BY 的“新”功能

7.4 GROUP BY 的CUBE 扩展

7.5 CUBE 的实际应用

7.6 通过GROUPING()函数排除空值

7.7 用GROUPING()来扩展报告

7.8 使用GROUPING_ID()来扩展报告

7.9 GROUPING SETS 与ROLLUP()

7.10 GROUP BY 局限性

7.11 小结

第8章 分析函数

8.1 示例数据

8.2 分析函数剖析

8.3 函数列表

8.4 聚合函数

8.4.1 跨越整个分区的聚合函数

8.4.2 细粒度窗口声明

8.4.3 默认窗口声明

8.5 Lead 和Lag

8.5.1 语法和排序

8.5.2 例1：从前一行中返回一个值

8.5.3 理解数据行的位移

8.5.4 例2：从下一行中返回一个值

8.6 First_value 和Last_value

8.6.1 例子：使用First_value 来计算最大值

8.6.2 例子：使用Last_value 来计算最小值

8.7 其他分析函数

8.7.1 Nth_value (11gR2)

8.7.2 Rank

8.7.3 Dense_rank

8.7.4 Row_number

8.7.5 Ratio_to_report

8.7.6 Percent_rank

8.7.7 Percentile_cont

8.7.8 Percentile_disc

8.7.9 NTILE

8.7.10 Stddev

8.7.11 Listagg

8.8 性能调优

8.8.1 执行计划

8.8.2 谓词

8.8.3 索引

8.9 高级话题

8.9.1 动态SQL

<<Oracle SQL高级编程>>

- 8.9.2 嵌套分析函数
- 8.9.3 并行
- 8.9.4 PGA 大小
- 8.10 组织行为
- 8.11 小结
- 第9章 Model 子句
- 9.1 电子表格
- 9.2 通过Model 子句进行跨行引用
 - 9.2.1 示例数据
 - 9.2.2 剖析Model 子句
 - 9.2.3 规则
- 9.3 位置和符号引用
 - 9.3.1 位置标记
 - 9.3.2 符号标记
 - 9.3.3 FOR 循环
- 9.4 返回更新后的行
- 9.5 求解顺序
 - 9.5.1 行求解顺序
 - 9.5.2 规则求解顺序
- 9.6 聚合
- 9.7 迭代
 - 9.7.1 一个例子
 - 9.7.2 PRESENTV与空值
- 9.8 查找表
- 9.9 空值
- 9.10 使用Model子句进行性能调优
 - 9.10.1 执行计划
 - 9.10.2 谓语句前推
 - 9.10.3 物化视图
 - 9.10.4 并行
 - 9.10.5 Model 子句执行中的分区
 - 9.10.6 索引
- 9.11 子查询因子化
- 9.12 小结
- 第10章 子查询因子化
- 10.1 标准用法
- 10.2 SQL 优化
 - 10.2.1 测试执行计划
 - 10.2.2 跨多个执行的测试
 - 10.2.3 测试查询改变的影响
 - 10.2.4 寻找其他优化机会
 - 10.2.5 将子查询因子化应用到PL/SQL 中
- 10.3 递归子查询
 - 10.3.1 一个CONNECT BY 的例子
 - 10.3.2 使用RSF 的例子
 - 10.3.3 RSF 的限制条件
 - 10.3.4 与CONNECT BY 的不同点

<<Oracle SQL高级编程>>

10.4 复制CONNECT BY 的功能

10.4.1 LEVEL 伪列

10.4.2 SYS_CONNECT_BY_PATH 函数

10.4.3 CONNECT_BY_ROOT 运算符

10.4.4 CONNECT_BY_ISCYCLE伪列和NOCYCLE 参数

10.4.5 CONNECT_BY_ISLEAF 伪列

10.5 小结

第11章 半联结和反联结

11.1 半联结

11.2 半联结执行计划

11.3 控制半联结执行计划

11.3.1 使用提示控制半联结执行计划

11.3.2 在实例级控制半联结执行计划

11.4 半联结限制条件

11.5 半联结必要条件

11.6 反联结

11.7 反联结执行计划

11.8 控制反联结执行计划

11.8.1 使用提示控制反联结执行计划

11.8.2 在实例级控制反联结执行计划

11.9 反联结限制条件

11.10 反联结必要条件

11.11 小结

第12章 索引

12.1 理解索引

12.1.1 什么时候使用索引

12.1.2 列的选择

12.1.3 空值问题

12.2 索引结构类型

12.2.1 B-树索引

12.2.2 位图索引

12.2.3 索引组织表

12.3 分区索引

12.3.1 局部索引

12.3.2 全局索引

12.3.3 散列分区与范围分区

12.4 与应用特点相匹配的解决方案

12.4.1 压缩索引

12.4.2 基于函数的索引

12.4.3 反转键索引

12.4.4 降序索引

12.5 管理问题的解决方案

12.5.1 不可见索引

12.5.2 虚拟索引

12.5.3 位图联结索引

12.6 小结

第13章 SELECT 以外的内容

<<Oracle SQL高级编程>>

13.1 INSERT

13.1.1 直接路径插入

13.1.2 多表插入

13.1.3 条件插入

13.1.4 DML 错误日志

13.2 UPDATE

13.3 DELETE

13.4 MERGE

13.4.1 语法和用法

13.4.2 性能比较

13.5 小结

第14章 事务处理

14.1 什么是事务

14.2 事务的ACID 属性

14.3 事务隔离级别

14.4 多版本读一致性

14.5 事务控制语句

14.5.1 Commit (提交)

14.5.2 Savepoint (保存点)

14.5.3 Rollback (回滚)

14.5.4 Set Transaction (设置事务)

14.5.5 Set Constraints (设置约束)

14.6 将运算分组为事务

14.7 订单录入模式

14.8 活动事务

14.9 使用保存点

14.10 序列化事务

14.11 隔离事务

14.12 自治事务

14.13 小结

第15章 测试与质量保证

15.1 测试用例

15.2 测试方法

15.3 单元测试

15.4 回归测试

15.5 模式修改

15.6 重复单元测试

15.7 执行计划比较

15.8 性能测量

15.9 在代码中加入性能测量

15.10 性能测试

15.11 破坏性测试

15.12 通过性能测量进行系统检修

15.13 小结

第16章 计划稳定性与控制

16.1 计划不稳定性：理解这个问题

16.1.1 统计信息的变化

<<Oracle SQL高级编程>>

- 16.1.2 运行环境的改变
- 16.1.3 SQL 语句的改变
- 16.1.4 绑定变量窥视
- 16.2 识别执行计划的不稳定性
 - 16.2.1 抓取当前所运行查询的数据
 - 16.2.2 查看一条语句的性能历史
 - 16.2.3 按照执行计划聚合统计信息
 - 16.2.4 寻找执行计划的统计方差
 - 16.2.5 在一个时间点附近检查偏差
- 16.3 执行计划控制：解决问题
 - 16.3.1 调整查询结构
 - 16.3.2 适当使用常量
 - 16.3.3 给优化器一些提示
- 16.4 执行计划控制：不能直接访问代码
 - 16.4.1 选项1：改变统计信息
 - 16.4.2 选项2：改变数据库参数
 - 16.4.3 选项3：增加或移除访问路径
 - 16.4.4 选项4：应用基于提示的执行计划控制机制
 - 16.4.5 大纲
 - 16.4.6 SQL 概要文件
 - 16.4.7 SQL 执行计划基线
 - 16.4.8 基于提示的执行计划控制机制总结
- 16.5 结论

章节摘录

版权页：插图：SQL的目的就是简单地提供一个到数据库的接口，在本书指的是Oracle数据库。

每一条SQL语句对于数据库来说就是一条命令或指令。

SQL与其他编程语言(如C或Java)的区别就在于它是要处理数据集合而不是一行一行的数据。

语言本身也不需要你提供如何导航到数据的指令——这是在后台透明地进行的。

但你将在后面的章节中看到，如果想在Oracle中写出高效的SQL语句，了解数据及其在数据库中的存储方式与存储位置是很重要的。

由于不同的供应商(例如甲骨文、IBM和微软)实现SQL核心功能的机制相差无几，所以基于某一种数据库所学的技巧同样可以应用到其他类型的数据库上。

你基本上可以利用同样的SQL语句来进行数据的查询、插入、更新和删除，以及创建、修改和删除对象，而不必管数据库的供应商是哪家。

尽管SQL是各种关系型数据库管理系统的标准语言，但实际上它并不一定是关系型的。

在本书后面我将就这一点稍作扩展。

如果想要了解更多的细节，我推荐大家阅读C.J.Date的SQL and Relational Theory一书。

需要铭记于心的一点是SQL语言并不总是严格遵守关系模型的——它根本就没有实现关系模型的某些要素，同时还不恰当地实现了一些要素。

事实上，既然SQL是基于关系模型的，那么要想写出尽可能正确高效的SQL语句，你不仅必须要理解SQL语言，还要理解关系模型。

<<Oracle SQL高级编程>>

媒体关注与评论

本书作者全部是OakTable的成员，且具有15 - 29年丰富的Oracle开发经验。在研究一些被其他专门讨论Oracle SQL语言的参考书直接忽略的问题时，这种对Oracle数据库的长期钻研无疑是一个巨大的优势。

——亚马逊读者评论

<<Oracle SQL高级编程>>

编辑推荐

《Oracle SQL高级编程》：资深Oracle专家力作，OakTable团队推荐，全面、独到、翔实，题材丰富，Oracle开发人员和DBA必备。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>