



Oracle 技术系列丛书

ORACLE®

AUTHORIZED ORACLE PRESS™—EXCLUSIVELY FROM OSBORNE

Oracle9i 升级指南

Oracle9i New Features

(美) Robert G. Freeman 著 袁勤勇 等译

1.13
8



OFFICIAL • AUTHORIZED

Oracle Press

ONLY FROM OSBORNE



机械工业出版社
China Machine Press



Education

Oracle技术系列丛书

Oracle9i 升级指南

(美) Robert G. Freeman 著

袁勤勇 等译



机械工业出版社
China Machine Press

本书详细介绍了Oracle公司新一代关系数据库Oracle9i的新特性。主要包括: Oracle9i的数据库管理特性, Oracle9i数据库体系结构的变化, Oracle9i可用性和可恢复性的新特性, 数据库DDS和数据仓库的新特性, Oracle9i SQL、PL/SQL的新特性, Oracle9i RAC等内容。本书内容实用, 叙述清晰, 实例丰富, 特别适合原来使用Oracle的其他版本现在要升级到Oracle9i的读者使用。Oracle DBA及其他Oracle的使用者使用本书可以迅速掌握Oracle9i数据库的特性。

Robert G.Freeman: Oracle9i New Features(ISBN:0-07-222385-5).

Original language copyright © 2002 by The McGraw-Hill Companies, Inc.

All rights reserved.

Simplified Chinese translation edition published by China Machine Press.

本书中文简体字版由美国麦格劳-希尔教育出版公司授权机械工业出版社出版, 未经出版者预先书面许可, 不得以任何方式复制或抄袭本书的任何部分。

本书封面贴有McGraw-Hill公司防伪标签, 无标签者不得销售。

版权所有, 侵权必究。

本书版权登记号: 图字: 01-2002-3607

图书在版编目(CIP)数据

Oracle9i 升级指南 / (美) 弗里曼 (Freeman, R. G.) 著; 袁勤勇等译. -北京: 机械工业出版社, 2003.3

(Oracle技术系列丛书)

书名原文: Oracle9i New Features

ISBN 7-111-11623-2

I. O… II. ①弗… ②袁… III. 关系数据库-数据库管理系统, Oracle9i IV. TP311.138

中国版本图书馆CIP数据核字(2003)第009753号

机械工业出版社(北京市西城区百万庄大街22号 邮政编码 100037)

责任编辑: 刘 晖 朱 劼

北京昌平奔腾印刷厂印刷·新华书店北京发行所发行

2003年3月第1版第1次印刷

787mm×1092mm 1/16·14.25印张

印数: 0 001-4 000册

定价: 29.00元

凡购本书, 如有倒页、脱页、缺页, 由本社发行部调换

前 言

2001年，Oracle发布了最新一代关系数据库系统（RDBMS）——Oracle9i。Oracle9i的特性丰富，它为数据库管理员提供了大量用来简化数据库管理、改进数据库性能、增强数据库可用性和可恢复性的新工具。本书用来帮助Oracle8i数据库管理员和开发人员迅速掌握Oracle9i的新特性。你将发现书中有很多的例子可帮助你快速将这些新特性用于工作中。希望我们的努力可以使你平稳升级到Oracle9i，从而成功地使用Oracle9i的新特性。

为了使本书能够及时出版，我们对大量的Oracle9i新特性进行了一定的选择。选择的原则是让读者用最少的努力获取最多的知识。所以，你可能会发现书中有些主题使用了大量篇幅来描述，而有的主题只是一笔带过，有的主题根本没有提及。由于受时间和篇幅的限制，我们必须对内容有所选择。希望你们觉得我们作出的选择是值得的，同时发现这本书是一本有价值的Oracle9i入门书。

如Oracle以前版本一样，Oracle9i还在不断发展。本书是基于Oracle9i的第1版撰写的。毫无疑问，随着Oracle9i新版本的面世，功能会有所改变，并且你会发现本书中的有些主题将不再准确。希望这本书能对你有所帮助，同时希望它能继续发挥指导性作用，并迎接新的Oracle关系数据库产品的出现。任何一本书的错误、问题都在所难免。你可通过电子邮件dbaoracle@aol.com与我们联系，提出修改意见或者作出（精辟）注解。如有问题请访问网站www.revealnet.com，我作为Quest Revealnet Labs DBA Pipeline的SYSQP成员在该网站上开设了专栏。你可以在网上提问并从DBA或开发者那里得到回答。最后，希望本书能帮助你迅速掌握Oracle9i的新特性。

目 录

前言	
第1章 Oracle9i 的数据库管理以及管理特性	1
1.1 兼容性参数	2
1.2 移植/升级提示	2
1.3 Oracle管理的数据文件	2
1.3.1 OMF的使用、规则和限制	3
1.3.2 配置数据库以使用OMF	7
1.3.3 使用OMF的一些实例	8
1.4 管理Oracle9i 的共享内存区域	11
1.4.1 支持多种数据库块大小	11
1.4.2 新的Oracle9i 内存数据库初始化参数	12
1.4.3 动态可变共享内存	14
1.4.4 自动PGA内存管理	15
1.5 Oracle9i 默认的临时表空间	16
1.6 Oracle9i 中的自动UNDO管理	17
1.6.1 创建UNDO表空间	17
1.6.2 配置使用自动UNDO管理的实例	19
1.6.3 数据字典和自动UNDO管理	20
1.7 可恢复的空间管理	21
1.7.1 可恢复的空间管理的特性和限制	21
1.7.2 可恢复的空间管理的候选数据库操作	22
1.7.3 对可恢复的空间管理特性的管理	23
1.7.4 可恢复的空间管理和Oracle工具	27
1.8 永久的初始化参数	28
1.8.1 创建服务器参数文件	28
1.8.2 设置永久化参数	29
1.8.3 管理spfile	30
1.9 Oracle 在9i 中支持的平台更少	31
第2章 Oracle9i 体系结构的变化	33
2.1 索引组织表的变化	34
2.1.1 索引组织表的索引选项	34
2.1.2 IOT、并行DML和溢出段	35
2.1.3 在IOT上重建B树二级索引	35
2.2 索引的跳过扫描	35
2.3 提取对象元数据	37
2.4 联机对象操作	39
2.4.1 索引的联机操作	39
2.4.2 联机移动IOT的溢出段	39
2.4.3 联机重组和重新定义表	40
2.5 自动段空间管理	46
2.6 位图连接索引	46
2.7 识别未使用的索引	48
2.8 游标共享	49
2.9 共享服务器改进	50
2.10 创建表空间默认值的变化	52
2.11 停顿数据库	52
第3章 Oracle9i 的可用性和可恢复性的新特性	55
3.1 Oracle9i flashback查询	56
3.1.1 什么是Oracle9i flashback 查询	56
3.1.2 设置数据库以应用Oracle9i flashback 查询	56
3.1.3 使用Oracle9i flashback查询特性	57
3.2 Oracle9i LogMiner的改进	60
3.2.1 转换与群集相关的DML语句	61
3.2.2 支持数据库DDL语句	61
3.2.3 从字典中提取重做日志或者使用联机数据字典	61
3.2.4 自动的字典升级和失效追踪	62
3.2.5 略过重做日志损坏的地方	63
3.2.6 只显示提交事务的功能	63
3.2.7 Oracle9i LogMiner限制	64

3.3 实例失败-恢复新特性	64	5.2 虚拟私有数据库的改进	120
3.3.1 减少崩溃和实例恢复时间	64	5.2.1 Oracle策略管理器	120
3.3.2 迅速启动基于时间的恢复	65	5.2.2 GAC	121
3.4 Oracle9i中的RMAN新特性	66	5.2.3 分区的细粒度访问控制	122
3.4.1 新的和变化了的管理特性	66	5.3 其他安全改进	123
3.4.2 新的和改变了的备份特性	70	5.3.1 标签安全	123
3.4.3 块介质恢复操作和可以重新启动的 恢复	74	5.3.2 DBMS_OBFUSCATION的改进	124
3.5 Oracle9i 数据防护	76	5.3.3 细粒度的审核	124
3.5.1 Oracle9i 数据防护结构和组件	76	5.3.4 Oracle数据库默认账户的安全	126
3.5.2 主数据库	77	5.4 NLS和全球化支持的改进	127
3.5.3 日志传输服务	78	5.4.1 字符与字节语义	127
3.5.4 Oracle9i 备用数据库: 角色管理和 出色的故障恢复	85	5.4.2 NCHAR数据类型的变化	128
3.5.5 Oracle9i 备用数据库和其他改进	86	5.4.3 改变数据库字符集	129
第4章 Oracle9i 数据库DDS和数据仓库的 新特性	89	5.4.4 时区和夏令时	129
4.1 Oracle9i列表分区	90	5.4.5 新的日期-时间数据类型	130
4.1.1 创建列表分区表	90	5.5 Oracle9i 配置程序的改进	133
4.1.2 列表分区表规则和特性	91	5.5.1 Oracle通用安装程序的改进	133
4.1.3 维护列表分区表	92	5.5.2 数据库配置助手的改进	133
4.2 视图约束	95	5.6 其他的Oracle9i 新特性和改进	134
4.2.1 引入视图约束	96	5.6.1 默认安装的高级复制	134
4.2.2 创建视图约束	96	5.6.2 把LONG转换成LOB数据类型	134
4.2.3 修改视图约束	97	5.6.3 数据库快速冻结	134
4.2.4 删除视图约束	98	5.7 Oracle9i 参数和视图的改变	135
4.3 Oracle9i ETL新特性	98	5.7.1 废弃的数据库参数	135
4.3.1 外部Oracle表	99	5.7.2 重命名参数	136
4.3.2 表函数	104	5.7.3 Oracle9i 中的新参数	136
4.4 其他新的、增强的数据仓库和ETL特性	107	5.7.4 Oracle9i 中新的数据字典视图	137
4.4.1 多表insert语句	107	5.7.5 Oracle9i 中重命名的数据字典视图	138
4.4.2 merge语句	110	5.7.6 Oracle9i 中新动态性能视图	138
4.4.3 Oracle数据变化捕捉	111	5.7.7 Oracle9i 中重命名的动态性能视图	139
4.4.4 Oracle9i并行直接加载改进	112	5.7.8 Oracle9i 中废弃的动态性能视图	140
4.4.5 Oracle9i 概要管理变化	113	第6章 Oracle9i SQL、PL/SQL新特性	141
第5章 Oracle9i 的其他特性及改进	119	6.1 新操作符: CROSS JOIN、NATURAL JOIN、USING、ON和JOIN	142
5.1 去除内部及服务器管理器之间的联系	120	6.2 新的outer join 操作符	142
		6.2.1 left outer join命令和 right outer join 命令	142

6.2.2 full outer join命令	143	第7章 Oracle9i RAC	169
6.3 case表达式的改进	144	7.1 新术语	170
6.3.1 简单型case语句	145	7.2 安装和配置	171
6.3.2 查找型case语句	146	7.3 Oracle9i 新的Cache Fusion实现	174
6.3.3 nullif 和 coalesce case语句	146	7.4 典型的ORAC块传输情形	177
6.3.4 标量子查询	147	7.5 HA特性	182
6.4 约束改进	149	7.5.1 HA的Cache Fusion	182
6.4.1 锁定和非索引外键	150	7.5.2 故障恢复和负载均衡	185
6.4.2 在FK创建期间存储主键	150	7.6 ORAC管理和诊断	190
6.4.3 与约束有关的索引管理特性	150	7.6.1 进程	190
6.4.4 在基于函数的索引上进行纯索引 扫描	151	7.6.2 新的实例控制命令	191
6.5 SQL新函数	151	7.6.3 数据字典对象	191
6.5.1 显式的默认列值	151	7.6.4 初始化参数	194
6.5.2 select for update命令中的wait子句	152	7.6.5 OEM 和 Oracle 配置助手	197
6.5.3 日期-时间新函数	152	7.6.6 SRVCTL工具	199
6.6 PL/SQL新函数	155	7.6.7 新的SQL脚本	200
6.6.1 通用SQL解析器	155	7.7 支持平台	201
6.6.2 case语句和表达式	155	7.7.1 Linux、Windows和 Veritas 群集 服务器	201
6.6.3 对LOB的无缝支持	157	7.7.2 UNIX	201
6.7 Oracle9i 对象改进	157	第8章 Oracle9i 中XML 和iFS新特性	203
6.7.1 继承	158	8.1 Oracle9i中的XML新特性	204
6.7.2 类型发展	158	8.2 Xpath的简要介绍	205
6.7.3 多层集合	158	8.2.1 XMLType 实例	208
6.8 批处理的改进	159	8.2.2 DBMS_XMLGEN	208
6.9 本地动态SQL的改进	160	8.2.3 SYS_XMLGEN	211
6.10 Oracle9i 中Internet 包的变化	161	8.2.4 SYS_XMLAGG	211
6.11 优化器和 Statistics-Gathering 新选项	162	8.2.5 URI 参考类型	212
6.11.1 first_rows优化新选项	162	8.2.6 XML开发工具箱	214
6.11.2 Statistics-Gathering 新选项	162	8.3 iFS新特性	215
6.11.3 新的优化器选项	163	8.3.1 管理特性	216
6.11.4 收集系统统计数据	164	8.3.2 协作特性	218
6.11.5 联机analyze...validate结构	167	8.3.3 开发特性	219
6.12 检查高速缓存的执行计划	167		



第1章 Oracle9i 的数据库管理 以及管理特性

- 兼容性参数
- Oracle管理的数据文件（OMF）
- 管理Oracle9i的共享内存区域
- Oracle9i默认的临时表空间
- Oracle9i中的UNDO表空间
- 可恢复的空间管理
- 永久的初始化参数
- Oracle9i支持的平台

欢迎你阅读本书第1章。本章我们开始学习Oracle9i新的和增强的特性。当然，数据库管理员们都希望这些特性可使Oracle数据库的管理更加简单迅速，并且尽量不要对用户产生影响。Oracle9i为实现这个目标提供了许多新特性，下面都将一一讲述。

1.1 兼容性参数

当我们审视Oracle9i众多新的、变化的、改进的特性时，首先想到了参数compatibility。它存在于数据库参数文件init.ora中，可以用来控制使用哪个数据库版本。这个参数在需要从其他版本移植到9i时非常有用，在成功移植到Oracle9i之前，最好保留原版本的compatibility参数。这是因为compatibility参数的值如果不是关于Oracle9i的，将不可使用Oracle9i的新特性。这使你很容易退回到以前的版本，因为你没有使用那些在退回到以前版本前必须删除的Oracle9i特性。

对于compatibility参数的不同值，某些数据库的默认行为（如创建表空间）也是不同的。一旦已经移植或升级到Oracle9i并熟悉了新的数据库，你就可以将compatibility参数的值设置为9.0.0.0，然后就可以使用本书介绍的Oracle9i的新特性了。

1.2 移植/升级提示

通常情况下，使用mig工具或数据库转化向导，所有的操作系统都支持Oracle 7.2或更新版本向Oracle9i的移植，很多平台甚至支持Oracle 7.1的移植。如果你运行的是Oracle8i，那么你需要使用平台特有的指令去升级数据库。这通常需要在Oracle9i环境下启动Oracle8i，并运行一个升级脚本。请参考你当前Oracle版本的文档，它为移植或升级提供了精确的指令。

1.3 Oracle管理的数据文件

在这一章里，你学到的第一个Oracle9i新特性是Oracle管理的数据文件（OMF）。OMF提供了管理数据库文件的能力。OMF作为Oracle9i的一部分，使数据库管理变得更容易。对于Oracle9i以前的版本，如果你要删除一个表空间，必须删掉与表空间相关的物理文件。但在Oracle9i中，你可通过OMF使物理文件的管理由数据库自行完成。在这一节里，将学到几种具有这种特性的数据文件，并可看到OMF优点和限制。然后，你将学到如何配置数据库，以便更好地利用OMF，以及一些使用OMF的实例。

1.3.1 OMF的使用、规则和限制

在这一节，你将学到与OMF相关的用法、规则和限制。首先，将看到OMF的概念以及何时应该使用，何时不应该使用。然后，我们将涉及OMF的管理。

1. OMF简介

当你需要创建数据库数据文件、临时文件、联机重做日志文件和数据库控制文件时，就会用到OMF。为了使用OMF，你必须将数据库设置为使用OMF的状态（参见1.3.2节）。数据库一旦设置完毕，那么在你没有明确定义与语句相关的数据文件时，Oracle 将在执行诸如create tablespace的DDL语句时，创建数据文件。在Oracle9i中OMF可与表空间、临时表空间、重做日志以及控制文件相关联。让我们看看OMF在创建表空间、重做日志以及控制文件时的特殊性。

表空间OMF 你可用OMF创建表空间，即使SYSTEM表空间也不例外。为了使Oracle具有这项功能，你必须在数据库参数文件中设置db_create_file_dest参数（参见1.3.2节）。例如，当你用create tablespace或 create temporary tablespace创建表空间，却没有提供数据文件名，Oracle就会自行为这个表空间创建数据文件。而且，如果你用create tablespace创建没有提供数据文件名的SYSTEM表空间，OMF数据文件也会被创建。还有，如果你用create tablespace创建DEFAULT表空间或UNDO表空间，那么系统也会为它们创建相应的OMF。任一OMF的默认大小是100M，而且数据文件可被设置为autoextend，其最大长度不受限制。

如果你想定义一个不是100M的数据文件，需要在命令中包括datafile关键字、size参数（没有数据文件名），那么数据文件即可按你要求的大小创建。你还可以通过autoextend off使数据文件不具有自动扩展的功能。请看下例：

```
CREATE TABLESPACE new_tbs DATAFILE SIZE 500M AUTOEXTEND OFF;
```

下面例子使用两个OMF创建表空间：

```
CREATE TABLESPACE new_tbs DATAFILE SIZE 500M, SIZE 500M AUTOEXTEND OFF;
```

你可以在不影响Oracle 数据库管理数据文件能力的情况下，改变数据文件的大小（通过alter database datafile resize命令）或改变数据文件的autoextend参数。

当数据文件充满了表空间时，只要数据库管理员不改变它的autoextend属性，它们将自动扩展。如果需要，数据库管理员除了可扩展已存在的数据文件外，还可选择使用alter tablespace

add datafile命令来创建额外的数据文件。如果数据库管理员未指定数据文件名，新添加的数据文件将是一个OMF。如果你愿意的话，你可将OMF和非OMF的数据文件在同一表空间里混合、匹配。除非你用drop tablespace命令中新的including contents and datafiles关键字，否则Oracle不会删除任何非OMF的数据文件。

当你删除一个包含OMF的表空间，Oracle将从操作系统中除去与表空间相关的所有OMF。例如，只要它们归Oracle管理，执行drop tablespace命令将导致Oracle删除与表空间相关的数据文件。当然，如果你已定义了数据文件的名字和位置，Oracle将不会删除这些文件。这需要数据库管理员自己负责删除。

另一个有趣的功能是：你可将人工定义的数据文件与OMF联系起来。例如，下面命令是完全合法的：

```
CREATE TABLESPACE new_tbs DATAFILE SIZE 500M,  
'd:\oracle\oradata\mydb\mydb_new_tbs_02.dbf' SIZE 500M AUTOEXTEND OFF;
```

在这个命令中，Oracle不但会创建OMF，还会创建人工定义的数据文件。如果要删除表空间，默认的方式是，Oracle只是自动删除OMF，而数据库管理员需手工删除所有的非Oracle管理的数据文件。这一特性可被扩展到所有包含人工创建的数据文件的表空间。例如，在现有的表空间中增加额外的OMF，可以扩展分配给表空间（原来在Oracle8i中创建，包含人工创建的数据文件）的空间。下面一节里，你将看到一些关于重做日志文件的OMF操作的例子。

重做日志OMF 如果你决定使用Oracle管理重做日志文件，你可根据需要在执行create database命令时，用maxlogfiles子句创建足够多的重做日志组。你可为每个组创建多达五个OMF成员（如果数据库已建成，用maxlogmembers设置）。如果用参数db_create_online_log_dest_n来定义多个组成员，那么不同的重做日志组成员将创建在不同的位置（参见1.3.2节，可以看到此类参数的列表）。

基于OMF重做日志，你可用create database命令新建数据库，就像本章后面的例子那样，你只需要忽略数据库数据文件名即可。

根据不同的操作系统，如果没有设置任何db_create_online_log_dest_n参数，那么系统将在db_create_file_dest参数指定的位置为每个重做日志文件组创建一个成员。在使用create database语句时，如果你没有设置任何一个参数，那么Oracle系统将报错。如果创建了OMF，并且使用alter database drop logfile group 或 alter database drop logfile member命令，Oracle将除去相关的

日志文件组或成员。默认情况下, Oracle管理的重做日志大小是100M。在这节后面, 你将看到几个重做日志文件的OMF操作的例子。

OMF控制文件 当你创建数据库时, 如果CONTROL_FILES参数没有列在数据库参数文件中, 并且create_online_log_dest_n被设置了, Oracle将为定义的目录创建OMF控制文件。就像重做日志文件一样, 可为即将创建的数据库配置多达5个控制文件的拷贝(参见1.3.2节)。如果设置了db_create_file_dest参数, 却没有设置db_create_online_log_dest_n, 那么将在db_create_file_dest指定的位置创建一个简单的控制文件。如果设置了db_create_online_log_dest_n, 那么控制文件将在该位置被写入。

根据不同的操作系统, 如果control_file、db_create_online_log_dest_n和db_create_file_dest参数都没有被设置, 那么Oracle可能会选择做一些几件事情。在某些情况下, Oracle可能在操作系统指定的目录下创建OMF控制文件, 此时, 这个控制文件将不是Oracle管理的控制文件。如果你希望控制文件由Oracle管理, 那么你必须保证OMF参数(包括db_create_file_dest和db_create_online_log_dest_n)被正确设置。对于某些平台, 如果控制文件的位置没有由control_file参数定义, Oracle只是简单的报告一个错误。

2. 何时使用OMF

Oracle OMF在很多情况下可用。首先, 它可用在少用户的小型数据库上, 来减少与之相关的数据库管理费用。这一特性减少了这种数据库需要的全部管理费用, 并可确保旧的无用数据文件不会降低磁盘空间的总体可用性。让数据库使用这种特性并不意味着定义数据库数据文件名和位置的功能已不可用, 事实上, 你可选择其中任何一种数据库功能。

OMF特性对于开发和测试数据库是十分有用的。使用这一特性, 可让开发者自如地去创建和删除他们自己的表空间(虽然, 我们在此并不支持强制使用OMF)。

OMF的另一个用处是简化备份数据库的管理。以前, 当我们对主数据库添加一个表空间或数据文件时, 必须人为介入备份数据库以执行同样的操作。现在, 用上OMF, 这种情形不再存在了。如果备份数据库被设置成使用OMF, 那么在主数据库创建表空间或添加数据文件的时候, 将会在备份服务器上自动创建表空间或数据文件, 不需要任何其他管理行为。

注意 如果主数据库中数据文件被删除, Oracle不会自动删除备份数据库中相关的数据文件。

如果你处在一个使用大型磁盘组的大规模数据库环境中, 可能会发现最好还是使用OMF。在这样的环境里, 通常创建少量大型文件系统, 并分割到几个磁盘上。这主要看你希望将它分割到几个磁盘, 如此可以显著地改进系统的性能。

对于没有使用高端磁盘分割列组的大容量或负责关键任务的数据库来说, OMF并不是恰当的选择。例如, 我们并不推荐在许多小型的文件系统或运行RAID-5的系统上使用OMF。这是因

为管理的数据文件的本质是这样的，DBA（数据库管理员）的某些任务，诸如IO分配，并不能得到这些特性真正的支持（在某种意义上说，是支持完成整体目标）。而且管理数据文件的特性并不支持原始磁盘设备的使用。

3. OMF管理

当Oracle创建被它管理的数据库数据文件的时候，它要遵循一个数据文件的命名规则。你不可以用Oracle命名OMF的规则来创建新的数据文件，任何这种企图都会导致一个错误。数据库数据文件的命名规则如下表所示。（注意，在不同的操作系统端口中可能有所不同，检查你的操作系统文档以得到正确的文件命名规则。）

文件类型	命名规则	实 例
数据文件	ora_{tablespace_name}_{unique character string}.dbf	ORA_NEW_TBS_ZV3NZF00.DBF
临时文件	ora_{tablespace_name}_{unique character string}.tmp	ORA_TEMP_TBS_ZV3NZF01.DBF
重做日志文件	ora_{online redo log group number}_{unique character string}.log	ORA_4_ZV307100.LOG
控制文件	ora_{unique character string}.ctl	ORA_4_ZV307100.CTL

注意 使用的表空间名最多有8个字符。这就是为什么表名的第二部分，即唯一性字符串很重要的原因。两个表空间可能会有不同的表名，但是表空间名的前8个字符可能是一样的。

作为一个数据库管理员，你可在SQL语句中像使用普通的数据文件名一样使用OMF名。例如，你可用alter database rename file 或 alter tablespace rename datafile命令重命名Oracle数据库管理的文件，也可用alter database drop logfile命令删除指定的Oracle管理的重做日志文件。

重命名一个OMF数据文件，首先需要将与OMF数据文件相关的表空间断开（或者断开OMF数据文件），然后在操作系统层上重命名数据文件名。一旦修改了数据文件名，你就可以用数据库的rename命令（使用alter database或者alter tablespace命令），重命名数据库中的OMF。最后，将表空间和数据文件相关联。

如果你在重命名OMF数据文件的时候使用的文件命名规则没有遵循OMF的文件命名规则，那么那个文件将不再是OMF。这样你不可以以“ORA_”开头来重命名任何非OMF Oracle数据文件。这将导致一个错误，ORA_前缀在Oracle中是OMF保留字符。

下面是一个对已存在的OMF数据文件重命名的例子：

```
alter session set db_create_file_dest = '/home1/teach3';
create tablespace sgdtest4 datafile size 2m;
select file_name,tablespace_name
from dba_data_files where tablespace_name = 'SDGTEST4';
```

```
FILE_NAME                                TABLESPACE_NAME
-----
/home1/teach3/ora_sdgtest4_xx5vcmqf.dbf  SDGTEST4

ALTER TABLESPACE sdgtest4 OFFLINE;

HOST ls
ora_sdgtest4_xx5vcmqf.dbf
host mv ora_sdgtest4_xx5vcmqf.dbf ora_sdgtest4_xx5vsdg4.dbf
alter tablespace sdgtest4 rename datafile
  '/home1/teach3/ora_sdgtest4_xx5vcmqf.dbf' to
  '/home1/teach3/ora_sdgtest4_xx5vsdg4.dbf';

ALTER TABLESPACE sdgtest4 ONLINE;
```

Oracle管理的数据库数据文件的备份和恢复过程与DBA管理的数据文件的相应过程没有什么不同，而且Oracle的imp和exp功能并不会因为OMF的存在而受到影响。使用控制文件备份恢复一个丢失的控制文件的过程，或者使用alter database backup control file to trace的结果来重新创建一个控制文件的过程仍然没有改变。

1.3.2 配置数据库以使用OMF

使用OMF，你必须首先设置一些数据库参数。这些参数定义了不同OMF的创建位置，与OMF相关的参数可在表1-1中看到。（在下一节你可看到一些例子。）

注意，在表中描述的每一个参数都可通过使用alter system 和 alter session命令动态地更改。例如：

```
ALTER SYSTEM SET DB_CREATE_FILE_DEST='d:\oracle\data\my_datafiles';
```

表1-1 数据库参数

参 数 名	默 认 值	作 用
db_create_file_dest	none	定义了存放OMF和临时文件的文件系统的位置。如果DB_CREATE_ONLINE_LOG_DEST_n参数没有被设置，这个位置也用于Oracle管理的控制文件和重做日志文件
db_create_online_log_dest_n	none	定义了将要创建的Oracle管理的联机重做日志文件的文件系统位置。值n可以是1~5的数字，它允许对每个联机重做日志组成员进行多达五次的多元拷贝，允许对创建的控制文件拷贝5次

改变创建文件的位置并不影响Oracle管理其他目录中已创建的数据文件的能力。

注意 即使你没有设置db_create_file_dest或db_create_online_log_dest_n 参数, 不需要关机 和重启数据库, 就可以利用OMF来动态设置它们。

1.3.3 使用OMF的一些实例

在这一节里, 你将可以学到一些使用OMF的不同数据库操作。首先, 你得配置一个参数文件, 以便你能使用OMF。然后, 你将学习用OMF为SYSTEM表空间、UNDO表空间、默认表空间、重做日志和控制文件创建数据库。接着, 我们将检查涉及OMF的几种不同类型操作的效果。这包括在数据库上删除表空间、添加删除联机重做日志。在这个例子中, 我们将使用我的叫mydb的Oracle9i 数据库。

注意 这些例子并不用作创建数据库的全部过程的一个指导, 在演示步骤前后还有几个步骤。这里假设你已熟悉了创建数据库的过程。

1. 将数据库设置成使用OMF状态

使用OMF的第一步是配置参数文件init.ora, 使之支持这一特性。下面是mydb数据库中init.ora的例子(我们已省去了许多与OMF配置无关的设置)。

```
db_name=mydb
undo_management=auto
undo_tablespace=myundotbs
DB_CREATE_FILE_DEST=c:\oracle\admin\mydb\data
DB_CREATE_ONLINE_LOG_DEST_1=c:\oracle\admin\mydb\redo
DB_CREATE_ONLINE_LOG_DEST_2=d:\oracle\admin\mydb\redo
```

在这例子里, mydb数据库被设置成自动撤销(undo)管理。我使用db_create_file_dest参数定义了一个叫MYUNDOTBS的UNDO表空间, 这有助于激活Oracle管理数据文件特性。在这种情况下, 如果Oracle数据库创建数据文件, 它将被放到c:\oracle\admin\mydb\data 目录下。

下面, 使用db_create_online_log_dest_1和db_create_online_log_dest_2参数为将要创建的重做日志和控制文件定义了两个不同的目录地址。注意, 为了保护重做日志和控制文件不被意外删除, 并保证IO平衡, 这两个文件被放在了不同的驱动器上。

2. 用OMF创建数据库

现在让我们来创建自己的数据库! 如果想让Oracle为我们完成所有的工作, 实际上我们只需要使用create database命令。在这种情况下, 如果OMF在init.ora中设置, Oracle将为你做好一切。然而, 为了更多地控制数据库创建过程, 让我们使用如下create database命令来创建mydb数据库:

```
CREATE DATABASE mydb
DATAFILE SIZE 200M
LOGFILE GROUP 1 SIZE 20M, GROUP 2 SIZE 20M
DEFAULT TEMPORARY TABLESPACE dflt_ts TEMPFILE SIZE 50M
UNDO TABLESPACE undo_ts DATAFILE SIZE 50M
MAXLOGFILES=5
MAXLOGMEMBERS=5
MAXDATAFILES=200
NOARCHIVELOG;
```

在我们的例子中，数据库mydb将被建为具有200M空间的SYSTEM表空间。根据我们在上一例里设置的参数，SYSTEM表空间的数据文件将被放在c:\oracle\admin\mydb\data目录下。接着，我们定义了一个叫dflt_ts的默认临时表空间，它有50M空间，也被放在c:\oracle\admin\mydb\data目录下。我们还创建了一个叫undo_ts的50M的UNDO表空间，这个表空间的数据文件仍将被放在c:\oracle\admin\mydb\data下。当这个数据库被创建时，重做日志文件将在两个位置被创建。每组第一个成员将被放在c:\oracle\admin\mydb\redo目录下，第二个成员将被放在d:\oracle\admin\mydb\redo目录下。

3. OMF的管理

现在，我们的数据库已被建立了。下面让我们看一些涉及OMF管理函数的例子，这里包括添加和删除表空间。你将学到的其他管理项是数据文件的默认位置，以及从使用OMF的数据库中添加和删除重做日志的过程。

添加表空间 当使用OMF时，添加表空间是一项简单的工作。简单到只需要使用一个带有表空间名字的create tablespace命令，并且Oracle将为即将创建的表空间默认分配100M数据文件空间，如下例所示：

```
CREATE TABLESPACE auto_created_tbs;
```

如果我们想创建更大的表空间，我们可在命令中包括DATAFILE子句，从而指出数据文件的大小。此外，在下面的例子中，通过在语句中包含NOEXTEND子句，数据文件的扩展能力将不可用。具体语句请看下例：

```
CREATE TABLESPACE bigger_tbs DATAFILE 200M NOEXTEND;
```

上一个语句创建了一个200M的数据文件，并禁用该文件的AUTOEXTEND功能。你还可以在创建临时表空间时使用OMF，如下例所示：

```
CREATE TEMPORARY TABLESPACE temp_obj_tbs DATAFILE 200M NOEXTEND;
```

删除表空间 既然我们已创建了表空间，我们有时需要删除它们。在这例子中，我们一起删除在这一节创建的bigger_tbs表空间。只需要使用drop tablespace命令，Oracle将处理其余所有过程。请看下例：

```
DROP TABLESPACE bigger_tbs;
```

在执行命令的过程中，bigger_tbs表空间的数据文件将被Oracle自动删除。

注意 即使db_create_file_dest参数已被改变，只要它保存在原来的目录中，OMF仍将被删除。

注意 drop tablespace语句的including contents子句已有了一个新的子句and datafiles。当这一子句被包含在drop tablespace命令中时，表空间以及所有相关的数据文件都将被删除。这是Oracle9i的一个新特性。

改变数据文件的创建位置 alter system 和 alter session命令可用来改变与OMF相关的参数。这样我们就可改变数据文件和重做日志在创建时候的位置。下面的例子改变参数db_create_file_dest设置的位置，然后创建数据文件。

```
ALTER SYSTEM SET DB_CREATE_FILE_DEST='d:\oracle\admin\mydb\data';  
CREATE TABLESPACE new_tbs DATAFILE SIZE 150m NOEXTEND;
```

添加重做日志组 当我们创建我们的数据库时，定义了两个200M的重做日志组。现在让我们创建第三个日志文件组。要做到这一点，我们只需使用如下命令：

```
ALTER DATABASE ADD LOGFILE;
```

你还可以使用如下命令：

```
ALTER DATABASE ADD LOGFILE GROUP 3 SIZE 300M;
```

值得一提的是，在上例中，重做日志组为300M，而不是默认的100M。

删除重做日志组 可能你已经发现，现有的重做日志成员并不够大，所以你想重新创建更大的。在这种情况下，首先你需要删除一个已存在的重做日志组，然后再重建它们。这是一个很简单的操作，执行alter database命令：

```
ALTER DATABASE DROP LOGFILE GROUP 1;
```

必须牢记，想添加一个额外的OMF重做日志组成员是不可能的（换句话说，不能使用alter