

**Oracle Database 11g:  
数据库管理 — 课堂练习 I**

**学生指南第 1 册**

D50102CN20

版本 2.0

2010 年 6 月

D66914

**ORACLE®**

## 作者

Deirdre Matishak

Mark Fuller

## 技术撰稿人和审稿人

Maria Billings

Herbert Bradbury

Yanti Chang

Timothy Chien

Andy Fotunak

Gerlinde Frenzen

Steve Friedberg

Joel Goodman

Vimala Jacob

Dominique Jeunot

Pete Jones

Fukue Kawabe

Donna Keesling

Sean Kim

Achiel Langers

Gwen Lazenby

Essi Parast

Randy Richeson

Joe Roch

Hilda Simon

Ira Singer

Jim Spiller

Supithran Thananayagam

Branislav Valny

Manju Varrier

## 编辑

Raj Kumar

Daniel Milne

## 制图员

Rajiv Chandrabhanu

## 出版商

Jobi Varghese

Veena Narasimhan

版权所有 © 2010, Oracle。保留所有权利。

## 免责声明

本文档包含专有权信息，并受版权法和其它知识产权法的保护。您可以复制和打印本文档以供在 Oracle 培训课程中单独使用。不得以任何方式修改或变更本文档。除了在依照版权法中制定的“合理使用”范围内使用本文档外，在未经 Oracle 明确授权的情况下，您不得以全部或部分的形式使用、共享、下载、上载、复制、打印、显示、展示、再版、发布、许可、张贴、传播或散布本文档。

本文档中包含的信息如有更改，恕不另行通知。如果您在本文档中发现任何问题，请书面通知：Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA。不能保证本文档中没有错误。

## 有限权利声明

如果将本文档交付给美国政府或代表美国政府使用本文档的任何人，请下列通知中的规定适用：

### U.S. GOVERNMENT RIGHTS

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

## 商标声明

Oracle 是 Oracle 公司和（或）其分公司的注册商标。其它名称可能是其各自拥有者的商标。

## 目录

### I 简介

- 课程目标 I-2
- 建议计划 I-3
- Oracle 产品和服务 I-4
- Oracle Database 11g: “g” 代表网格 I-5
- 用于单实例的 Oracle Grid Infrastructure I-7

### 1 了解 Oracle DB 体系结构

- 课程目标 1-2
- Oracle DB 1-3
- 连接到服务器 1-4
- Oracle DB 服务器体系结构: 概览 1-6
- 实例: 数据库配置 1-7
- 连接到数据库实例 1-8
- Oracle DB 内存结构 1-9
- 共享池 1-11
- 数据库缓冲区高速缓存 1-13
- 重做日志缓冲区 1-14
- 大型池 1-15
- Java 池和流池 1-16
- 程序全局区 (PGA) 1-17
- 小测验 1-18
- 进程体系结构 1-20
- 进程结构 1-21
- 数据库写进程 (DBWn) 1-23
- 日志写进程 (LGWR) 1-25
- 检查点进程 (CKPT) 1-27
- 系统监视器进程 (SMON) 1-28
- 进程监视器进程 (PMON) 1-29
- 恢复器进程 1-30
- 归档进程 (ARCn) 1-31

进程启动顺序 1-32  
数据库存储体系结构 1-33  
逻辑和物理数据库结构 1-35  
段、区和块 1-37  
表空间和数据文件 1-38  
SYSTEM 和 SYSAUX 表空间 1-39  
自动存储管理 1-40  
ASM 存储组件 1-41  
与 Oracle DB 交互：内存、进程和存储 1-42  
小测验 1-44  
小结 1-46  
练习 1：概览 1-47

## 2 安装 Oracle 软件

课程目标 2-2  
Oracle DB 管理员的任务 2-3  
用于管理 Oracle DB 的工具 2-4  
制定安装计划 2-6  
Oracle Grid Infrastructure 和 Oracle DB 安装：系统要求 2-7  
准备操作系统 2-8  
设置环境变量 2-9  
检查系统要求 2-11  
Oracle Universal Installer (OUI) 2-12  
示例：安装场景 2-13  
第一部分：安装用于独立服务器的 Oracle Grid Infrastructure 2-14  
选择产品语言 2-15  
创建 ASM 磁盘组 2-16  
定义 ASM 口令 2-17  
定义已授权的操作系统组 2-18  
指定安装位置 2-19  
创建清单 2-20  
执行先决条件检查 2-21  
验证安装概要数据 2-22  
监视安装进度 2-23  
执行 root 配置脚本 2-24  
执行 Configuration Assistant 2-25

完成安装	2-26
配置 FRA 磁盘组	2-27
小测验	2-28
第二部分：安装 Oracle DB 软件	2-30
选择安装类型	2-31
选择网格安装选项	2-32
选择语言设置	2-33
选择数据库版本	2-34
指定安装位置	2-35
选择操作系统组	2-36
执行先决条件检查	2-37
安装概要页	2-38
安装产品页	2-39
安装完成	2-40
安装选项：无提示模式	2-41
小测验	2-42
小结	2-44
练习 2 概览：准备数据库环境	2-45

### 3 使用 DBCA 创建 Oracle DB

课程目标	3-2
计划数据库	3-3
数据库：示例	3-4
选择适当的字符集	3-5
如何使用字符集	3-7
要避免的问题	3-8
Database Configuration Assistant (DBCA)	3-9
使用 DBCA 创建数据库	3-10
创建数据库概要	3-16
口令管理	3-17
创建数据库设计模板	3-18
使用 DBCA 删除数据库	3-19
使用 DBCA 完成其它任务	3-21
小测验	3-22
小结	3-24
练习 3 概览：使用 DBCA	3-25

## 4 管理数据库实例

- 课程目标 4-2
- 管理框架 4-3
- 启动和停止 Database Control 4-4
- Oracle Enterprise Manager 4-5
  - “Database (数据库) ” 主页 4-7
- 其它 Oracle 工具 4-8
- 使用 SQL\*Plus 4-9
- 从 Shell 脚本调用 SQL\*Plus 4-10
- 从 SQL\*Plus 调用 SQL 脚本 4-11
- 初始化参数文件 4-12
- 简化初始化参数 4-14
- 初始化参数: 示例 4-15
- 使用 SQL\*Plus 查看参数 4-19
- 更改初始化参数值 4-21
- 更改参数值: 示例 4-23
- 小测验 4-24
- 数据库启动和关闭: 身份证明 4-26
- 启动 Oracle DB 实例 4-27
- 启动 Oracle DB 实例: NOMOUNT 4-28
- 启动 Oracle DB 实例: MOUNT 4-29
- 启动 Oracle DB 实例: OPEN 4-30
- 启动选项: 示例 4-31
- 关闭 Oracle DB 实例 4-32
- 关闭模式 4-33
- 关闭选项 4-34
- 关闭选项: 示例 4-37
- 查看预警日志 4-38
- 使用跟踪文件 4-40
- 动态性能视图 4-42
- 动态性能视图: 用法示例 4-43
- 动态性能视图: 注意事项 4-44
- 数据字典: 概览 4-45
- 数据字典视图 4-46
- 数据字典: 用法示例 4-48

小测验 4-49  
小结 4-51  
练习 4 概览：管理 Oracle 实例 4-52

## 5 管理 ASM 实例

课程目标 5-2  
ASM 对于管理员的好处 5-3  
ASM 实例 5-4  
ASM 组件：ASM 实例 — 主要进程 5-6  
ASM 实例初始化参数 5-7  
数据库实例与 ASM 之间的交互 5-9  
ASM 实例：动态性能视图 5-10  
ASM 系统权限 5-11  
使用 Oracle Enterprise Manager 管理 ASM 用户 5-12  
启动和停止 ASM 实例使用 SQL\*Plus 5-13  
启动和停止 ASM 实例使用 srvctl 5-15  
启动和停止 ASM 实例使用 asmcmd 5-16  
磁盘组概览 5-17  
ASM 磁盘 5-18  
分配单元 5-19  
ASM 文件 5-20  
区映射 5-21  
条带化粒度 5-22  
细粒度条带化 5-23  
ASM 故障组 5-25  
条带化和镜像示例 5-26  
故障示例 5-27  
管理磁盘组 5-28  
创建和删除磁盘组使用 SQL\*Plus 5-29  
向磁盘组添加磁盘 5-30  
其它 ALTER 命令 5-31  
使用 Oracle Enterprise Manager 管理 ASM 5-32  
ASM 磁盘组兼容性 5-33  
ASM 磁盘组属性 5-35

使用 Oracle Enterprise Manager 编辑磁盘组属性 5-36  
检索 ASM 元数据 5-37  
ASM 快速镜像重新同步概览 5-38  
小测验 5-39  
小结 5-41  
练习 5 概览：管理 ASM 实例 5-42

## 6 配置 Oracle Network 环境

课程目标 6-2  
Oracle Net 服务 6-3  
Oracle Net 监听程序 6-4  
建立网络连接 6-5  
建立连接 6-6  
用户会话 6-7  
配置和管理 Oracle Network 的工具 6-8  
监听程序控制实用程序 6-10  
监听程序控制实用程序的语法 6-11  
使用 SRVCTL 启动和停止监听程序 6-13  
监听程序主页 6-14  
“Net Services Administration（网络服务管理）”页 6-15  
创建监听程序 6-16  
添加监听程序地址 6-17  
数据库服务注册 6-18  
命名方法 6-20  
简便连接 6-21  
本地命名 6-22  
目录命名 6-23  
外部命名方法 6-24  
配置服务别名 6-25  
高级连接选项 6-26  
测试 Oracle Net 连接 6-28  
用户会话：专用服务器进程 6-29  
用户会话：共享服务器进程 6-30  
SGA 和 PGA 6-31

共享服务器：连接共享	6-32
不能使用共享服务器的情况	6-33
配置数据库之间的通信	6-34
连接到其它数据库	6-35
小测验	6-36
小结	6-38
练习 6 概览：使用 Oracle Network 组件	6-39

## 7 管理数据库存储结构

课程目标	7-2
表数据的存储方式	7-3
数据库块：内容	7-4
浏览存储结构	7-5
创建新的表空间	7-6
表空间的存储	7-10
预配置的数据库中的表空间	7-12
变更表空间	7-14
表空间操作	7-16
删除表空间	7-18
查看表空间信息	7-19
查看表空间内容	7-20
Oracle 管理的文件 (OMF)	7-22
扩大数据库	7-24
小测验	7-25
小结	7-27
练习 7 概览：管理数据库存储结构	7-28

## 8 管理用户安全性

课程目标	8-2
数据库用户帐户	8-3
预定义管理帐户	8-5
创建用户	8-6
验证用户	8-7
管理员验证	8-9
解除用户帐户的锁定并重置口令	8-10
权限	8-11

系统权限 8-12  
对象权限 8-14  
撤销带 ADMIN OPTION 的系统权限 8-15  
撤销带 GRANT OPTION 的对象权限 8-16  
角色的优点 8-17  
将权限分配给角色以及将角色分配给用户 8-18  
预定义角色 8-19  
创建角色 8-20  
保护角色 8-21  
将角色分配给用户 8-22  
小测验 8-23  
概要文件和用户 8-25  
实施口令安全功能 8-27  
创建口令概要文件 8-29  
提供的口令验证函数: VERIFY\_FUNCTION\_11G 8-30  
将限额分配给用户 8-31  
应用最少权限原则 8-33  
保护授权帐户 8-35  
小测验 8-36  
小结 8-38  
练习 8 概览: 管理用户 8-39

## 9 管理数据并发处理

课程目标 9-2  
锁 9-3  
锁定机制 9-4  
数据并发处理 9-5  
DML 锁 9-7  
入队机制 9-8  
锁冲突 9-9  
锁冲突的可能原因 9-10  
检测锁冲突 9-11  
解决锁冲突 9-12  
使用 SQL 解决锁冲突 9-13  
死锁 9-14

- 小测验 9-15
- 小结 9-17
- 练习 9 概览：管理数据和并发处理 9-18

## 10 管理还原数据

- 课程目标 10-2
- 还原数据 10-3
- 事务处理和还原数据 10-5
- 存储还原信息 10-6
- 还原数据与重做数据 10-7
- 管理还原 10-8
- 配置还原保留时间 10-9
- 保证还原保留时间 10-11
- 将还原表空间改为固定大小 10-12
- 一般还原信息 10-13
- 使用还原指导 10-14
- 查看系统活动 10-15
- 小测验 10-16
- 小结 10-18
- 练习 10 概览：管理还原段 10-19

## 11 实施 Oracle DB 审计

- 课程目标 11-2
- 责任分离 11-3
- 数据库安全性 11-4
- 监视合规性 11-6
- 标准数据库审计 11-7
- 配置审计线索 11-8
- 统一审计线索 11-9
- 指定审计选项 11-10
- 默认审计 11-11
- Enterprise Manager 审计页 11-12
- 使用和维护审计信息 11-13

- 基于值审计 11-14
- 细粒度审计 11-16
- FGA 策略 11-17
- 审计的 DML 语句: 注意事项 11-19
- FGA 准则 11-20
- SYSDBA 审计 11-21
- 维护审计线索 11-22
- Oracle Audit Vault 11-23
- 小测验 11-24
- 小结 11-26
- 练习 11 概览: 实施 Oracle DB 安全性 11-27

## **12 数据库维护**

- 课程目标 12-2
- 数据库维护 12-3
- 查看预警历史记录 12-4
- 术语 12-5
- Oracle 优化程序: 概览 12-6
- 优化程序统计信息 12-7
- 使用“管理优化程序统计信息”页 12-8
- 手动搜集优化程序统计信息 12-9
- 用于搜集统计信息的首选项 12-11
- 自动工作量资料档案库 (AWR) 12-13
- AWR 基础结构 12-14
- AWR 基线 12-15
- Enterprise Manager 和 AWR 12-16
- 管理 AWR 12-17
- 统计级别 12-18
- 自动数据库诊断监视器 (ADDM) 12-19
- ADDM 查找结果 12-20
- ADDM 建议 12-21
- 指导框架 12-22
- Enterprise Manager 和指导 12-24
- DBMS ADVISED 程序包 12-25
- 小测验 12-26

自动维护任务	12-27
自动维护任务配置	12-29
服务器生成的预警	12-30
设置阈值	12-32
创建和测试预警	12-33
预警通知	12-34
对预警作出响应	12-36
预警类型和清空预警	12-37
小测验	12-38
小结	12-39
练习 12 概览：预先维护	12-40

## 13 性能管理

课程目标	13-2
性能监视	13-3
Enterprise Manager 的性能页	13-4
细化到特定的等待类别	13-5
性能页：吞吐量	13-6
性能监视：顶级会话	13-7
性能监视：顶级服务	13-9
管理内存组件	13-10
启用自动内存管理 (AMM)	13-11
启用自动共享内存管理 (ASMM)	13-13
自动共享内存指导	13-14
动态性能统计信息	13-15
故障排除和优化视图	13-17
无效和不可用对象	13-18
小测验	13-20
小结	13-22
练习 13 概览：监视和改进性能	13-23

## 14 备份和恢复的概念

- 课程目标 14-2
- 部分工作内容 14-3
- 故障类别 14-5
- 语句失败 14-6
- 用户进程失败 14-7
- 网络故障 14-8
- 用户错误 14-9
- 闪回技术 14-10
- 实例故障 14-11
- 了解实例恢复：检查点 (CKPT) 进程 14-12
- 了解实例恢复：重做日志文件和日志写进程 14-13
- 了解实例恢复 14-14
- 实例恢复的阶段 14-15
- 优化实例恢复 14-16
- 使用 MTTR 指导 14-17
- 介质故障 14-18
- 配置可恢复性 14-19
- 配置快速恢复区 14-20
- 多路复用控制文件 14-21
- 重做日志文件 14-23
- 多路复用重做日志 14-24
- 归档日志文件 14-26
- 归档 (ARC $n$ ) 进程 14-27
- 归档日志文件：命名与目标位置 14-28
- 启用 ARCHIVELOG 模式 14-30
- 小测验 14-31
- 小结 14-33
- 练习 14 概览：配置可恢复性 14-34

## 15 执行数据库备份

- 课程目标 15-2
- 备份解决方案：概览 15-3
- Oracle Secure Backup 15-4
- 用户管理的备份 15-5
- 术语 15-6
- Recovery Manager (RMAN) 15-8
- 配置备份设置 15-9
- 调度备份：策略 15-11
- 调度备份：选项 15-12
- 调度备份：设置 15-13
- 调度备份：调度 15-14
- 调度备份：复查 15-15
- 将控制文件备份到跟踪文件 15-16
- 管理备份 15-17
- 查看备份报告 15-18
- 监视快速恢复区 15-19
- 使用 RMAN 命令行 15-21
- 小测验 15-22
- 小结 15-23
- 练习 15 概览：创建数据库备份 15-24

## 16 执行数据库恢复

- 课程目标 16-2
- 打开数据库 16-3
- 使数据库保持在打开状态 16-5
- Data Recovery Advisor（数据恢复指导） 16-6
- 丢失了控制文件 16-8
- 丢失了重做日志文件 16-9
- 在 NOARCHIVELOG 模式下丢失了数据文件 16-11
- 在 ARCHIVELOG 模式下丢失了非关键数据文件 16-12
- 在 ARCHIVELOG 模式下丢失了系统关键数据文件 16-13
- 数据故障：示例 16-14
- 数据恢复指导 16-15
- 评估数据故障 16-16

- 数据故障 16-17
- 列出数据故障 16-18
- 提供修复建议 16-19
- 执行修复 16-20
- 数据恢复指导视图 16-21
- 小测验 16-22
- 小结 16-24
- 练习 16 概览：执行数据库恢复 16-25

## 17 移动数据

- 课程目标 17-2
- 移动数据：一般体系结构 17-3
- Oracle 数据泵：概览 17-4
- Oracle 数据泵：优点 17-5
- 数据泵的目录对象 17-7
- 创建目录对象 17-8
- 数据泵导出与导入客户机：概览 17-9
- 数据泵实用程序：界面与模式 17-10
- 使用 Database Control 进行数据泵导出 17-11
- 数据泵导出示例：基本选项 17-12
- 数据泵导出示例：高级选项 17-13
- 数据泵导出示例：文件 17-14
- 数据泵导出示例：调度 17-16
- 数据泵导出示例：复查 17-17
- 数据泵导入示例：impdp 17-18
- 数据泵导入：转换 17-19
- 使用 Oracle Enterprise Manager 监视数据泵作业 17-20
- 以数据泵旧模式提供移植支持 17-21
- 数据泵旧模式 17-22
- 管理文件位置 17-25
- SQL\*Loader：概览 17-26
- 使用 SQL\*Loader 加载数据 17-28
- SQL\*Loader 控制文件 17-29
- 加载方法 17-31
- 外部表 17-32
- 外部表的优点 17-33

使用 ORACLE_LOADER 定义外部表	17-34
使用 ORACLE_DATAPUMP 填充外部表	17-35
使用外部表	17-36
数据字典	17-37
小测验	17-38
小结	17-40
练习 17 概览：移动数据	17-41

## 18 使用技术支持

课程目标	18-2
使用支持工作台	18-3
在 Oracle Enterprise Manager 中查看严重错误预警	18-4
查看问题详细资料	18-5
查看意外事件详细资料：转储文件	18-6
查看意外事件详细资料：检查器查找结果	18-7
创建服务请求	18-8
将诊断数据打包并上载到 Oracle 技术支持	18-9
跟踪服务请求并实施修复	18-10
关闭意外事件和问题	18-12
意外事件打包配置	18-13
Enterprise Manager 针对 ASM 的支持工作台	18-15
使用 Oracle 技术支持	18-16
My Oracle Support 集成	18-17
使用 My Oracle Support	18-18
调查问题	18-20
记录服务请求	18-22
管理补丁程序	18-24
应用补丁程序版本	18-25
使用补丁程序指导	18-26
使用补丁程序向导	18-27
应用补丁程序	18-28
存放补丁程序	18-29
联机打补丁：概览	18-30
安装联机补丁程序	18-31
联机打补丁的优点	18-32

- 常规打补丁和联机打补丁 18-33
- 联机打补丁注意事项 18-34
- 小测验 18-36
- 小结 18-37
- 练习 18 概览：对预警和补丁程序使用 EM 工具 18-38

#### 附录 A: 练习和解答

#### 附录 B: 基本的 Linux 和 vi 命令

#### 附录 C: SQL 语句语法

#### 附录 D: Oracle 后台进程

#### 附录 E: 缩写词和术语

#### 附录 F: Oracle Restart

- 课程目标 F-2
- Oracle Restart F-3
- Oracle Restart 进程的启动 F-5
- 控制 Oracle Restart F-6
- 选择正确的 SRVCTL 实用程序 F-8
- Oracle Restart 配置 F-9
- 使用 SRVCTL 实用程序 F-10
- 获取有关 SRVCTL 实用程序的帮助 F-11
- 使用 SRVCTL 实用程序启动组件 F-12
- 使用 SRVCTL 实用程序停止组件 F-13
- 查看组件状态 F-14
- 显示组件的 Oracle Restart 配置 F-15
- 手动向 Oracle Restart 配置添加组件 F-16
- 小测验 F-17
- 小结 F-18
- 练习 3-1: 概览 F-19

## 附录 G: 继续学习和深入阅读

- 您想从本课程中学到什么 G-2
- 继续学习资源 G-3
- Oracle University G-4
- 继续学习 G-5
- 数据库专业领域 G-6
- Oracle Real Application Clusters G-7
- Oracle Data Guard G-8
- 流概览 G-9
- Oracle 技术网 G-11
- 安全性 G-12
- Oracle by Example G-13
- Oracle 杂志 G-14
- Oracle 应用产品社区 G-15
- 技术支持: My Oracle Support G-16
- Oracle DB 产品页 G-17
- 感谢 G-18



# I 简介

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课程后，应能完成以下工作：

- 安装、创建和管理 Oracle Database 11g 发行版 2
- 配置应用程序的数据库
- 使用基本监视过程
- 实施备份和恢复策略
- 在数据库和文件之间移动数据

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 课程目标

在本课程中，您将安装 Oracle Database 11g 发行版 2 企业版软件、创建新数据库并学习如何管理数据库。

还可以学习通过配置数据库来支持应用程序和执行创建用户、定义存储结构和设置安全性等任务。本课程使用一个虚构应用程序。但是，执行的全部核心任务都是实际应用程序需要执行的任务。

配置数据库之后数据库管理工作并没有结束。还要学习如何通过设计备份和恢复策略来保护数据库，以及如何通过监视数据库来确保数据库顺利运行。

## 建议计划

天数	课程	天数	课程
1	1. 了解 Oracle DB 体系结构 2. 准备数据库环境 3. 创建 Oracle DB 4. 管理数据库实例	3	9. 管理数据并发处理 10. 管理还原数据 11. 实施 Oracle DB 审计
2	5. 管理 ASM 实例 6. 配置 Oracle Network 环境 7. 管理数据库存储结构 8. 管理用户安全性		12. 数据库维护 13. 性能管理 14. 备份和恢复的概念
		5	15. 执行数据库备份 16. 执行数据库恢复 17. 移动数据 18. 使用技术支持



版权所有 © 2010, Oracle。保留所有权利。

## Oracle 产品和服务

- Oracle DB
- Oracle WebLogic Application Server
- Oracle Applications
- Oracle Collaboration Suite
- Oracle Developer Suite
- Oracle 服务



**ORACLE®**

ORACLE

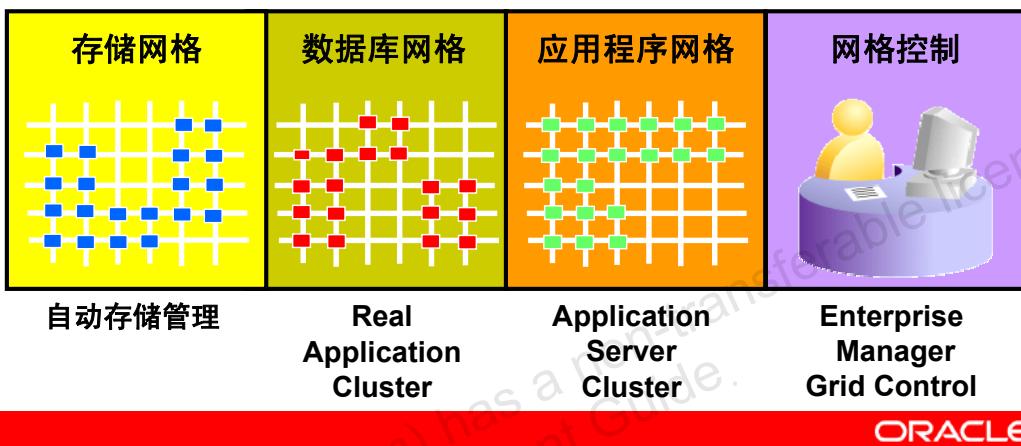
版权所有 © 2010, Oracle。保留所有权利。

### Oracle 产品和服务

- **Oracle DB:** Oracle DB 是为进行企业网格计算而设计的第一款数据库，这是一种最灵活、具有成本效益的管理信息和应用程序的方式。
- **Oracle WebLogic Application Server:** 是 Oracle 获得 Java 2 平台企业版认证的服务器，这种服务器中集成了开发和部署基于 Web 的应用程序所需的全部组件。它可用于部署电子商务门户、Web 服务和事务处理应用程序，如 PL/SQL、Oracle Forms 和基于 Java EE 的应用程序。
- **Oracle Applications:** Oracle E-Business Suite 是一套完整的商务应用程序，可用于管理和自动处理整个组织内的流程。
- **Oracle Collaboration Suite:** Oracle Collaboration Suite 是一个集成式的系统，可用来处理组织中的所有通信数据，其中包括：语音、电子邮件、传真、无线数据、日历信息和文件。
- **Oracle Developer Suite:** Oracle Developer Suite 是一个完整的集成环境，其中组合了应用程序开发工具与业务智能工具。
- **Oracle 服务:** Oracle 服务，如 Oracle Consulting 和 Oracle University，可以提供关于 Oracle 项目的必要专业知识。有关各种资源的链接，请参阅附录“下一步：继续学习”。

## Oracle Database 11g: “g” 代表网格

- 开放网格论坛 (Open Grid Forum, OGF)
- Oracle 网格基础结构:
  - 低成本
  - 高服务质量
  - 易于管理



版权所有 © 2010, Oracle。保留所有权利。

### Oracle Database 11g: “g” 代表网格

开放网格论坛 (OGF) 是一个负责制定网格计算标准的标准团体。该团体由一些委员会和工作小组构成，他们致力于制定网格计算各方面的标准。这些委员会和工作小组由来自学术界、研究团体和（日益增加的）商业公司的参与者组成。您可以访问 OGF 网站 <http://www.ogf.org>。

Oracle 创建的网格计算基础结构软件可在多个服务器中平衡各种类型的工作量，并将所有这些服务器作为一个完整的系统来管理。因为所有组件都通过集群方式集中在一起，所以网格计算可以获得与大型机计算一样高的可靠性。但是，与大型机和大型 UNIX 对称多处理 (SMP) 服务器不同，网格的构建使用开放系统技术，如 Intel 处理器和 Linux 操作系统，因此成本很低。

Oracle 网格计算技术包括：

- 自动存储管理 (ASM)
- Real Application Cluster (RAC)
- Application Server Cluster
- Enterprise Manager Grid Control

## Oracle Database 11g: “g” 代表网格（续）

**自动存储管理:** 该技术在所有磁盘间分布数据库数据，创建存储网格并进行维护，并以极低的管理成本提供极高的输入/输出 (I/O) 吞吐量。添加或删除磁盘时，ASM 会自动重新分布数据。（不需要使用逻辑卷管理器来管理文件系统。）由于可以选择使用镜像，数据可用性提高了，可以联机添加或删除磁盘。请参阅“管理数据库存储结构”一课。

**Oracle Real Application Cluster:** 在服务器集群上运行，可调整集群中所有应用程序的工作量，同时还提供以下功能：

- **集成式集群件:** 其中包括用于进行集群连接、消息传送和锁定、集群控制与恢复的功能。在 Oracle Database 10g 或更高版本支持的所有平台上均可使用这些功能。
- **自动工作量管理:** 可以定义一些规则，以便在正常操作和对故障作出响应期间将处理资源自动分配给每个服务。用户可以动态地修改这些规则，以满足不断变化的业务需求。这种在数据库网格中动态分配资源的功能是 Oracle RAC 特有的功能。
- **向中间层自动发送事件通知:** 集群配置发生更改后，中间层可以立即适应实例故障转移或可用的新实例。这样，最终用户在发生实例故障转移时可继续工作，不存在通常由网络超时引起的延迟。有新实例可用时，中间层可以立即开始与该实例建立负载平衡连接。Oracle Database 10g 或更高版本中的 Java 数据库连接 (JDBC) 驱动程序拥有“快速连接故障转移”功能，该功能可以自动启用以处理以上事件。

**Oracle WebLogic Application Grid:** 可以用于任何应用程序服务器（包括 Oracle WebLogic Server、IBM WebSphere Application Server 和 JBoss Application Server），也可以在不包含应用程序服务器的纯网格环境中使用。Oracle WebLogic Application Grid 可提供极高并且可预测的应用程序可伸缩性和性能。由于容量可以根据需要增加，Oracle WebLogic Application Grid 可以线性扩展中间件基础结构，将其从几台服务器扩展到数千台。通过内存中数据网格解决方案，Oracle WebLogic Application Grid 可提供对常用数据的快速访问。利用这种网格功能，数据计算可以并行进行，从而进一步提高应用程序的性能。

**Enterprise Manager Grid Control:** 用于管理网格级操作，包括管理整个软件堆栈、预配用户、克隆数据库和管理补丁程序。它可从最终用户的角度监视所有应用程序的性能。Grid Control 将网格基础结构的性能和可用性视为一个统一的整体，而不是独立的存储单元、数据库和应用程序服务器。硬件节点、数据库和应用程序服务器可以组成单个的逻辑实体，因而可将一组目标作为一个单元来管理。

注：本课程中，您将使用 Enterprise Manager Database Console 一次管理一个数据库。

## 用于单实例的 Oracle Grid Infrastructure

用于单实例的 Oracle Grid Infrastructure 是在 Oracle Database 11g 发行版 2 中引入的。

- 使用集群件介质进行安装，与 Oracle DB 软件彼此独立
- 包含 Oracle 自动存储管理 (ASM)
- 包含 Oracle Restart (一种针对非集群数据库的高可用性解决方案)
  - 可监视和重新启动以下组件：
    - 数据库实例
    - Oracle Net 监听程序
    - 数据库服务
    - 自动存储管理 (ASM) 实例
    - ASM 磁盘组
    - 用于 Data Guard 的 Oracle 通知服务 (ONS/eONS)



版权所有 © 2010, Oracle。保留所有权利。

### 用于单实例的 Oracle Grid Infrastructure

用于单实例的 Oracle Grid Infrastructure 是在 Oracle Database 11g 发行版 2 中引入的。它使用集群件介质进行安装，与 Oracle DB 软件彼此独立，现在包含 Oracle 自动存储管理以及称为 Oracle Restart 的新功能。

Oracle Restart 是为提高 Oracle DB 的可用性而设计的。它是一项仅针对单实例（非集群）环境的高可用性解决方案。对于 Oracle Real Application Cluster (Oracle RAC) 环境，自动重新启动组件的功能是由 Oracle Clusterware 提供的。Oracle Restart 可监视以下组件的健康状况并根据需要自动重新启动相应的组件：

- 数据库实例
- Oracle Net 监听程序
- 数据库服务
- ASM 实例
- ASM 磁盘组
- 用于 Data Guard 的 Oracle 通知服务 (ONS/eONS)

Oracle Restart 可以确保按照组件依赖关系以正确顺序启动组件。如果必须要关闭某个组件，其将确保首先彻底关闭从属组件。可以从 Oracle Grid Infrastructure 主目录运行 Oracle Restart，该目录与 Oracle DB 主目录是分开安装的。



# 了解 Oracle DB 体系结构

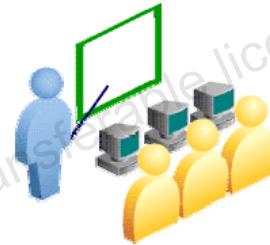
版权所有 © 2010, Oracle。保留所有权利。

ORACLE

## 课程目标

学完本课后，应能完成以下工作：

- 列出 Oracle DB 的主要体系结构组件
- 说明内存结构
- 描述后台进程
- 将逻辑存储结构与物理存储结构关联起来
- 描述 ASM 存储组件



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 课程目标

本课将详细地综述 Oracle DB 的体系结构。您将了解其物理和逻辑结构以及各种组件。

## Oracle DB

Oracle 关系数据库管理系统 (RDBMS) 提供了开放的、全面的、集成的信息管理方法



ORACLE

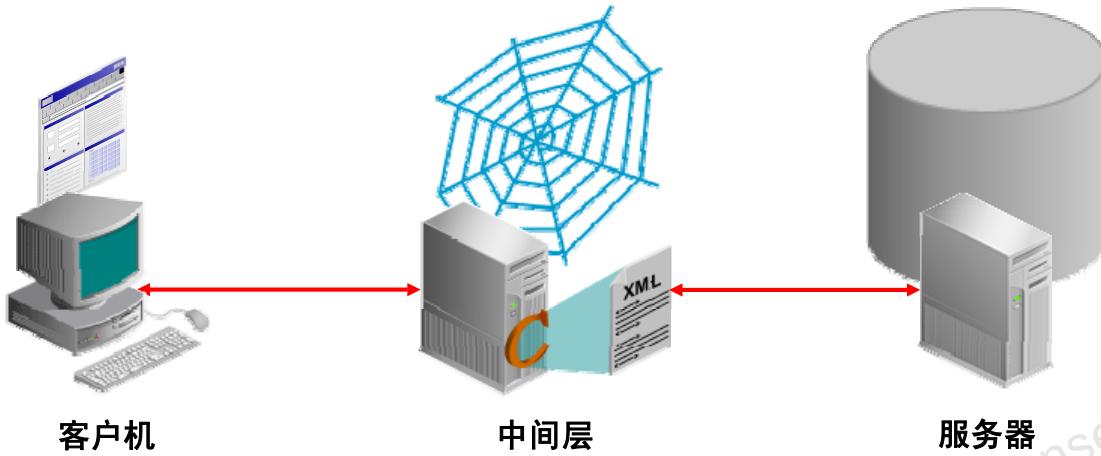
版权所有 © 2010, Oracle。保留所有权利。

### Oracle DB

数据库是被视为单元的数据集合，用于存储和检索相关信息。

Oracle 关系数据库管理系统 (RDBMS) 的可靠性非常高，可以管理多用户环境下的大量数据，这样很多用户可以并行访问同一数据。与此同时，还可以保持较高的性能。此外，它可以阻止未授权的访问，并为故障恢复提供有效的解决方案。

## 连接到服务器



所示为多层体系结构

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 连接到服务器

数据库用户可以使用以下三种方法之一连接到 Oracle 服务器：

- 用户登录到运行 Oracle 实例的操作系统，然后启动一个访问该系统上数据库的应用程序或工具。使用主机操作系统中可用的进程间通信机制建立通信路径。
- 用户启动本地计算机上的应用程序或工具，并通过网络连接到运行 Oracle DB 的计算机。在这种配置（称为“客户机/服务器”）中，使用网络软件在用户和后端服务器之间进行通信。客户机/服务器体系结构数据库系统具有两个部分：一个前（客户机）和一个后端（服务器），它们通过网络连接在一起。网络软件用于在用户和 Oracle 服务器之间进行通信。
  - 客户机是一个数据库应用程序，它启动一个请求，以便在数据库服务器上执行操作。它请求、处理并显示由服务器管理的数据。客户机工作站可以针对其作业进行优化。例如，客户机可能不需要大磁盘容量，或者可能从图形功能中获益。通常，客户机在数据库服务器以外的计算机上运行。多个客户机可以同时基于一个服务器运行。
  - 服务器运行 Oracle DB 软件并处理并行的共享数据访问所需的功能。服务器接收并处理源自客户机应用程序的请求。用于管理服务器的计算机可以针对其职责进行优化。例如，服务器计算机可以具有大容量磁盘和高速处理器。

## 连接到服务器（续）

- 用户通过本地计算机（客户机）上的工具（例如 Web 浏览器）访问应用程序服务器。然后，应用程序服务器代表客户机与后端数据库服务器进行交互。

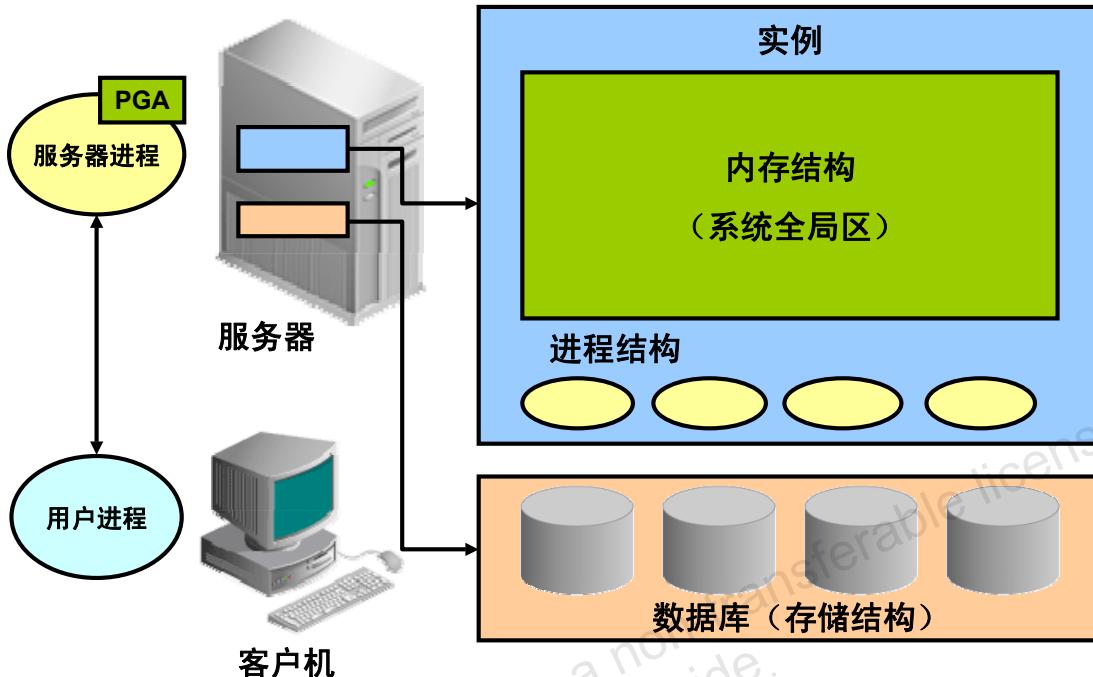
传统的多层体系结构具有以下组件：

- 启动操作的客户机或启动程序进程。
- 一个或多个执行部分操作的应用程序服务器。应用程序服务器包含大部分应用程序逻辑，为客户机提供数据访问，并执行一些查询处理，从而卸下了数据库服务器的一些负载。应用程序服务器可以充当客户机和多个数据库服务器之间的接口，并可以提供更高级别的安全性。
- 用于存储操作中使用的大多数数据的后端服务器或数据库服务器。

该体系结构允许使用应用程序服务器来执行以下任务：

- 验证客户机（例如 Web 浏览器）的身份证明。
- 连接到 Oracle DB 服务器。
- 代表客户机执行请求的操作。

## Oracle DB 服务器体系结构：概览



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle DB 服务器体系结构

Oracle DB 服务器体系结构包含以下三种主要结构：内存结构、进程结构和存储结构。

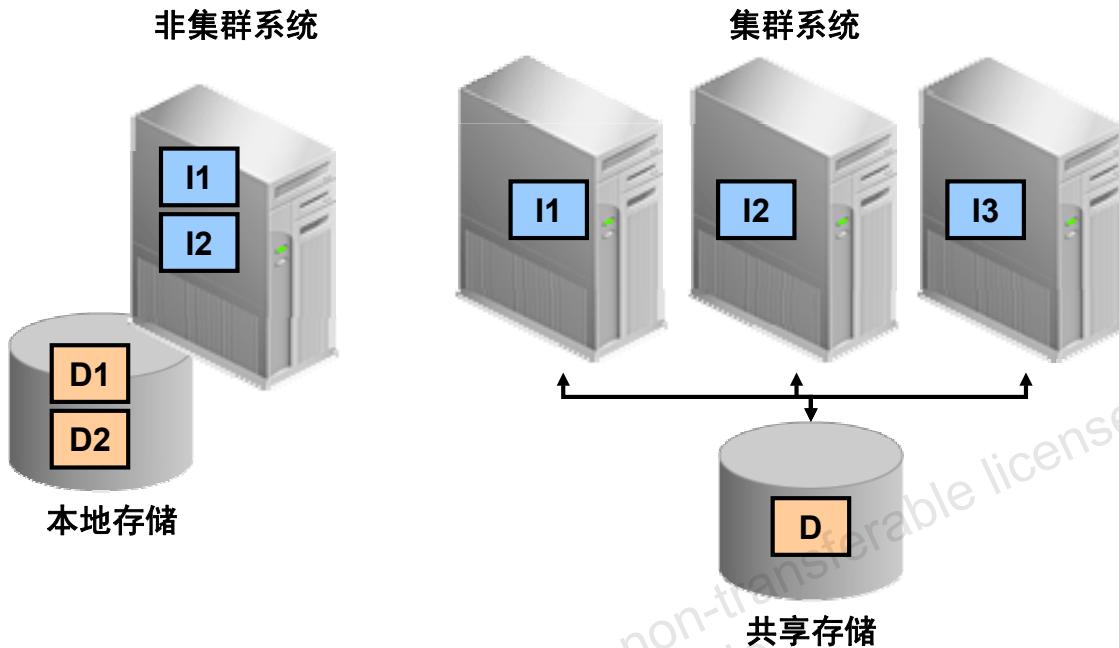
基本的 Oracle DB 系统由 Oracle DB 和数据库实例组成。

数据库包括物理结构和逻辑结构。由于物理结构和逻辑结构是分开的，因此管理数据的物理存储时不会影响对逻辑存储结构的访问。

实例由与该实例关联的内存结构和后台进程构成。每当启动一个实例时，都会分配一个称为系统全局区 (SGA) 的共享内存区，并启动后台进程。进程是在计算机的内存中运行的作业。进程被定义为操作系统中可运行一系列步骤的“控制线程”或机制。启动数据库实例后，Oracle 软件会将该实例与特定数据库相关联。该操作称为“装载数据库”。之后用户可以打开数据库，即授权用户可以对其进行访问。

**注：**Oracle 自动存储管理 (ASM) 在管理内存和进程组件时使用实例概念，不与特定数据库关联。

## 实例：数据库配置



ORACLE

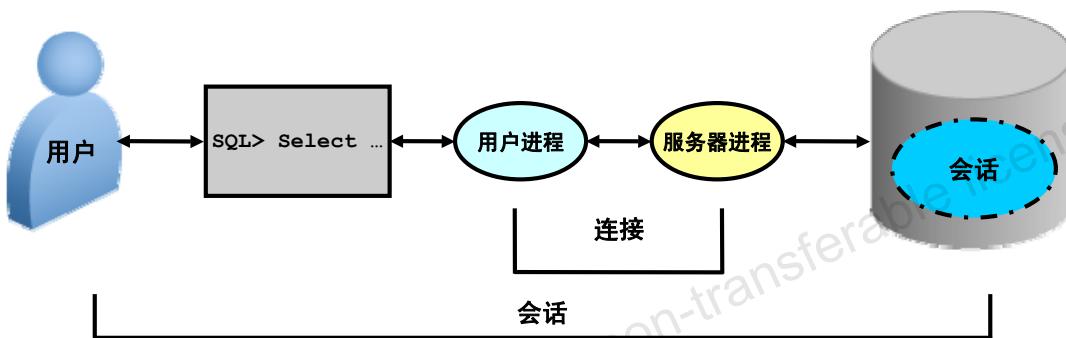
版权所有 © 2010, Oracle。保留所有权利。

## 实例：数据库配置

每个数据库实例都与一个并且只与一个数据库关联。如果同一服务器上有多个数据库，那么每个数据库会有一个单独的数据库实例。数据库实例不能共享。Oracle Real Applications Cluster (RAC) 数据库通常在几个独立服务器上运行共享同一数据库的多个实例。在此模式下，每个 RAC 实例都与同一数据库相关联，从而满足了最多只能有一个数据库与一个实例关联的要求。

## 连接到数据库实例

- 连接：用户进程和实例之间的通信
- 会话：用户通过用户进程与实例之间建立的特定连接



版权所有 © 2010, Oracle。保留所有权利。

ORACLE

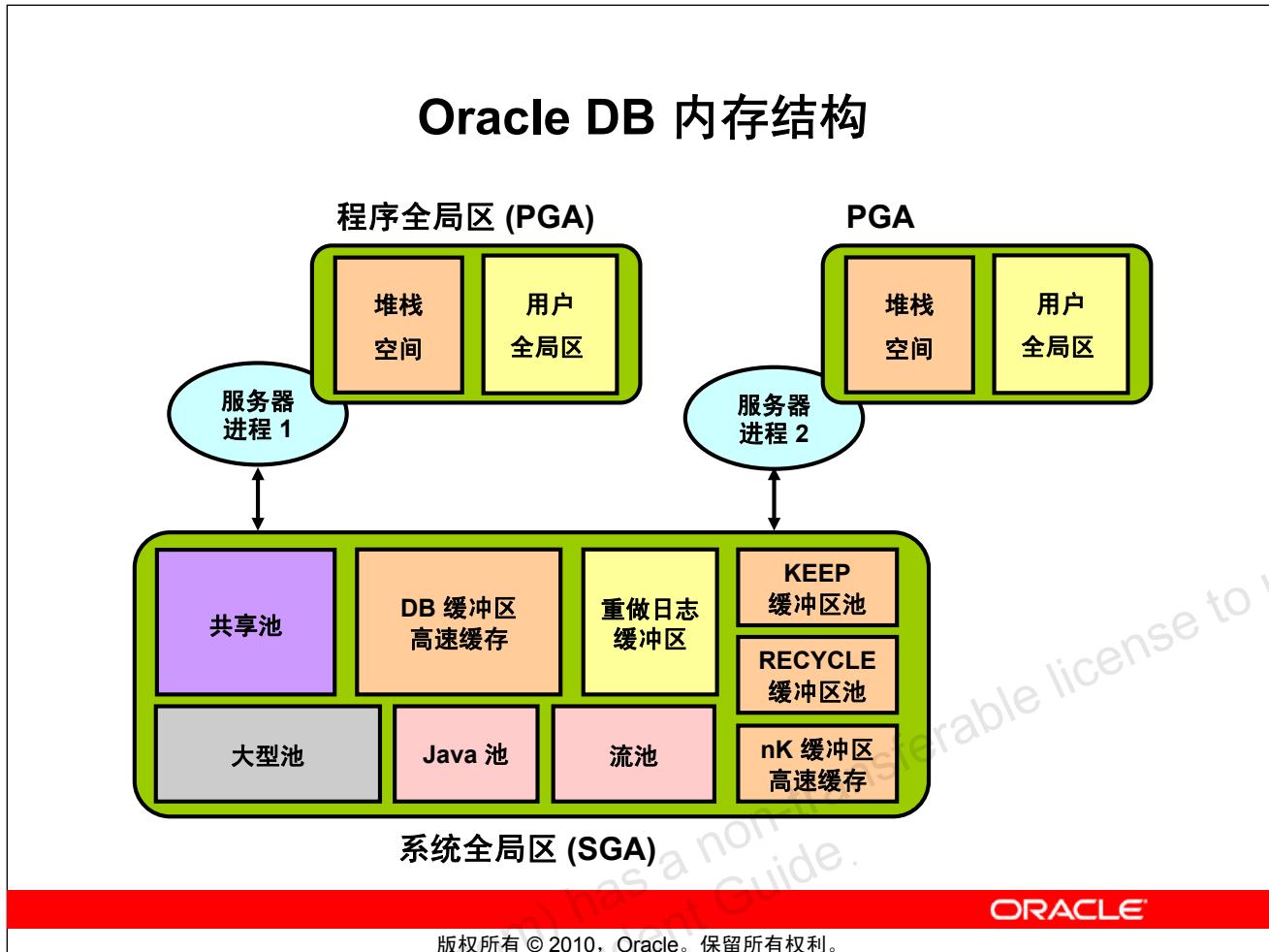
### 连接到数据库实例

连接和会话都与用户进程紧密相关，但在含义上有很大的区别。

“连接”是用户进程和 Oracle DB 实例之间的通信路径。通信路径是使用可用的进程间通信机制（用户进程和 Oracle DB 在同一计算机上运行）或网络软件（数据库应用程序和 Oracle DB 在不同计算机上运行并通过网络进行通信）建立的。

“会话”代表登录到数据库实例的当前用户的状态。例如，当某个用户启动 SQL\*Plus 时，该用户必须提供有效的用户名和口令，然后系统会为该用户建立一个会话。会话从用户连接时开始，一直持续到用户断开连接或退出数据库应用程序。

一个 Oracle DB 用户可以使用同一用户名创建多个会话，这些会话可以并存。例如，用户名/口令为 HR/HR 的用户可以多次连接到同一个 Oracle DB 实例。



## Oracle DB 内存结构

Oracle DB 创建并使用内存结构来满足多种需要。例如，使用内存来存储正在运行的程序代码、在各用户之间共享的数据以及所连接的每个用户的专用数据区域。

一个实例有两个关联的基本内存结构：

- **系统全局区 (SGA):** 一组共享的内存结构（称为 SGA 组件），其中包含一个 Oracle DB 实例的数据和控制信息。SGA 由所有服务器进程和后台进程共享。SGA 中存储的数据有高速缓存的数据块和共享 SQL 区域等。
- **程序全局区 (PGA):** 包含某个服务器进程或后台进程的数据及控制信息的内存区域。PGA 是 Oracle DB 在服务器进程或后台进程启动时创建的非共享内存。服务器进程对 PGA 的访问是独占式的。每个服务器进程和后台进程都具有自己的 PGA。

## Oracle DB 内存结构（续）

SGA 是包含实例数据和控制信息的内存区。SGA 包含以下数据结构：

- **共享池：**用于缓存可在用户间共享的各种构造
- **数据库缓冲区高速缓存：**用于缓存从数据库中检索到的数据块
- **KEEP 缓冲区池：**一种专用数据库缓冲区高速缓存，用于长时间在内存中保留数据块
- **RECYCLE 缓冲区池：**一种专用数据库缓冲区高速缓存，用于从内存中快速回收或删除数据块
- **nK 缓冲区高速缓存：**多种专用数据库缓冲区高速缓存中的一种，用于存放大小不同于默认数据库块大小的数据块
- **重做日志缓冲区：**重做信息（用于实例恢复）在写入磁盘中存储的物理重做日志文件之前，将缓存在此处
- **大型池：**可选区域，用于为某些大型进程（例如 Oracle 备份和恢复操作）和 I/O 服务器进程提供大型内存分配
- **Java 池：**用于存储 Java 虚拟机 (JVM) 中特定于会话的所有 Java 代码和数据
- **流池：**供 Oracle Streams 用来存储捕获和应用操作所需的信息

使用 Oracle Enterprise Manager 或 SQL\*Plus 启动实例时，会显示为 SGA 分配的内存量。

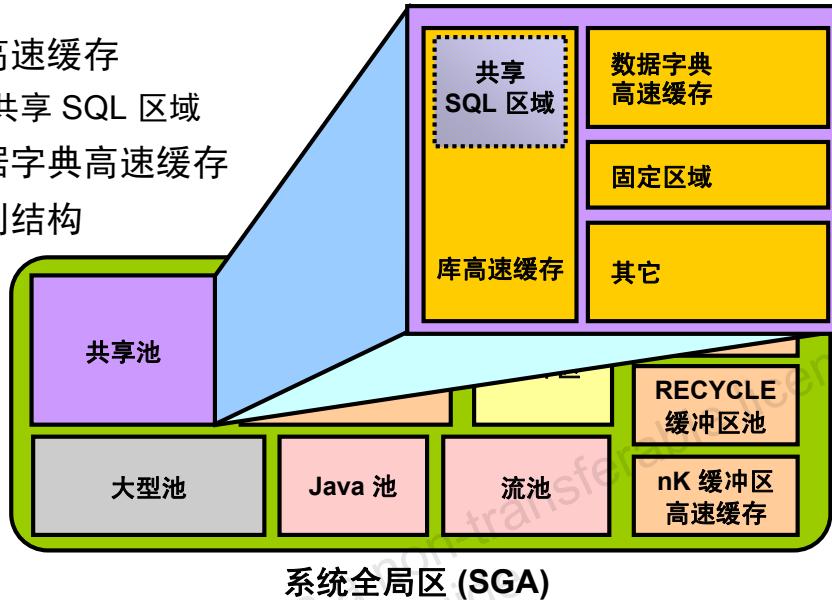
程序全局区 (PGA) 是一个内存区，其中包含每个服务器进程的数据及控制信息。Oracle 服务器进程为客户机请求提供服务。每个服务器进程都有自己的专用 PGA，该区域是在服务器进程启动时分配的。PGA 只能由相应的服务器进程访问，并且只有代表该服务器进程的 Oracle 代码可对其进行读写。PGA 分成两个主要区：堆栈空间和用户全局区 (UGA)。

使用动态 SGA 基础结构，可以在不关闭实例的情况下更改数据库缓冲区高速缓存、共享池、大型池、Java 池和流池的大小。

Oracle DB 使用初始化参数来创建和管理内存结构。管理内存的最简单方法是允许数据库自动管理和优化内存。在大多数平台上，您只需设置目标内存大小初始化参数 (MEMORY\_TARGET) 和最大内存大小初始化参数 (MEMORY\_MAX\_TARGET)，就可以做到这一点。

## 共享池

- 属于 SGA
- 包含：
  - 库高速缓存
    - 共享 SQL 区域
  - 数据字典高速缓存
  - 控制结构



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 共享池

SGA 的共享池部分包含库高速缓存、数据字典高速缓存、SQL 查询结果高速缓存、PL/SQL 函数结果高速缓存、并行执行消息的缓冲区以及控制结构。

“数据字典”是数据库表和视图的集合，其中包含有关数据库、数据库结构及其用户的参考信息。在 SQL 语句语法分析期间，Oracle DB 会频繁访问数据字典。该访问操作对于 Oracle DB 的持续操作来说至关重要。

Oracle DB 对数据字典的访问十分频繁，因此在内存中指定了两个特殊的位置来存放字典数据。一个区域称为“数据字典高速缓存”，也称为行高速缓存，因为它以行的形式存放数据，而不是以缓冲区的形式存放数据（缓冲区用于存放完整的数据块）。内存中的另一个用于存放字典数据的区域称为“库高速缓存”。所有 Oracle DB 用户进程都共享这两个高速缓存以便访问数据字典信息。

Oracle DB 使用共享 SQL 区域（以及 PGA 中保留的专用 SQL 区域）来表示它所运行的每个 SQL 语句。Oracle DB 可识别两个用户执行相同 SQL 语句的情况，从而为这些用户重用共享 SQL 区域。

共享 SQL 区域包含给定 SQL 语句的语法分析树和执行计划。Oracle DB 通过为多次运行的 SQL 语句使用一个共享 SQL 区域来节省内存。当许多用户运行同一个应用程序时，同一 SQL 语句通常会多次运行。

## 共享池（续）

对新的 SQL 语句进行语法分析时，Oracle DB 会从共享池中分配内存，以便在共享 SQL 区域中存储语句。该内存的大小取决于语句的复杂性。

Oracle DB 处理 PL/SQL 程序单元（过程、函数、程序包、匿名块和数据触发器）的方式与它处理单个 SQL 语句的方式十分相似。Oracle DB 分配一个共享区域以存放程序单元在经过语法分析和编译之后的形式。Oracle DB 分配一个专用区域，以便存放运行程序单元的会话所特有的值，包括局部变量、全局变量和程序包变量（也称为程序包实例化），并分配用于执行 SQL 的缓冲区。如果多个用户运行同一个程序单元，则所有用户都使用同一个共享区域，但其各自的专用 SQL 区域都维护着一个单独的副本，用于存放特定于自身会话的值。

PL/SQL 程序单元中包含的单个 SQL 语句的处理方式与其它 SQL 语句的处理方式类似。无论这些 SQL 语句在 PL/SQL 程序单元中的来源如何，它们都使用共享区域来存放其语法分析表示，并且为运行语句的每个会话使用一个专用区域。

SQL 查询结果高速缓存和 PL/SQL 函数结果高速缓存是 Oracle Database 11g 中新增的功能。它们共享相同的基础结构，出现在相同的动态性能 (V\$) 视图中，并且使用所提供的同一程序包进行管理。

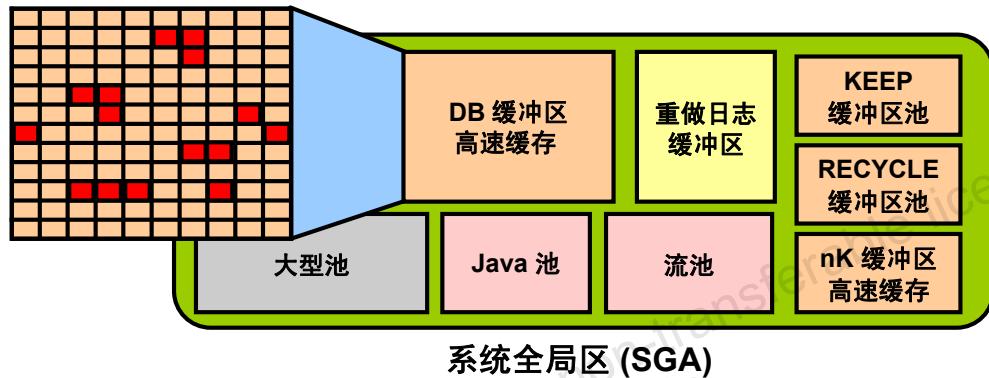
查询的结果和查询片段的结果可以高速缓存在 SQL 查询结果高速缓存的内存中。这样在将来执行这些查询和查询片段时，数据库可以使用高速缓存结果进行响应。由于从 SQL 查询结果高速缓存中检索结果要比重新运行查询快得多，因此将频繁运行的查询的结果缓存起来可以大大提高这些查询的性能。

如果计算的输入是 PL/SQL 函数发出的一个或若干个参数化查询，则有时会使用该函数来返回计算结果。在某些情况下，这些查询访问的数据很少更改（与调用函数的频率相比）。您可以在 PL/SQL 函数的源文本中包括语法，以请求将函数结果缓存在 PL/SQL 函数结果高速缓存中，并在有 DML 操作处理表列表中的表时清除高速缓存（以确保正确性）。

共享池的固定区域表示 SGA 的启动开销。与常见大小的共享池或 SGA 相比，它非常小。

## 数据库缓冲区高速缓存

- 属于 SGA
- 存放从数据文件读取的数据块的副本
- 由所有并行用户共享



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据库缓冲区高速缓存

数据库缓冲区高速缓存是 SGA 的一部分，用于存放从数据文件中读取的块映像或动态构造的块映像，以便符合读一致性模型要求。并行连接到实例的所有用户共享对数据库缓冲区高速缓存的访问。

Oracle DB 用户进程第一次请求特定数据片段时，将在数据库缓冲区高速缓存中搜索数据。如果该进程在高速缓存中找到数据（称为高速缓存命中），则直接从内存中读取数据。如果进程在高速缓存中找不到数据（称为高速缓存未命中），则在访问数据之前，必须将磁盘上的数据文件中的数据块复制到高速缓存中的缓冲区中。高速缓存命中时访问数据的速度要比高速缓存未命中时快。

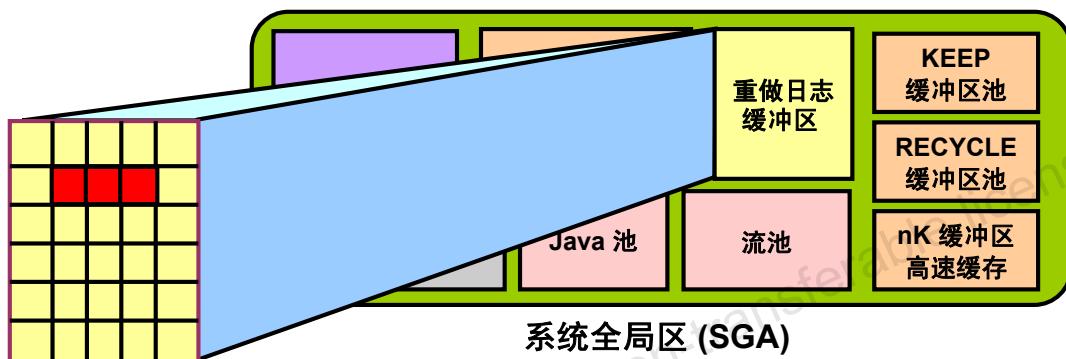
高速缓存中的缓冲区由一个复杂算法管理，该算法组合使用最近最少使用 (LRU) 列表和停靠计数。LRU 有助于确保最近使用的块往往都留在内存中，从而最大限度地减少磁盘访问。

KEEP 缓冲区池和 RECYCLE 缓冲区池用于优化专用缓冲区池。KEEP 缓冲区池用于延长缓冲区在内存中的保留时间，使其比使用 LRU 时的正常保留时间长。RECYCLE 缓冲区池用于加快内存中缓冲区的刷新速度，使其比使用 LRU 时的正常刷新速度快。

可以配置更多缓冲区高速缓存，以用于存放大小与默认块大小不同的块。

## 重做日志缓冲区

- 是 SGA 中的循环缓冲区
- 存放有关对数据库所做更改的信息
- 包含重做条目，这些条目中包含对 DML 和 DDL 等操作所做的更改进行重做的相关信息



版权所有 © 2010, Oracle。保留所有权利。

### 重做日志缓冲区

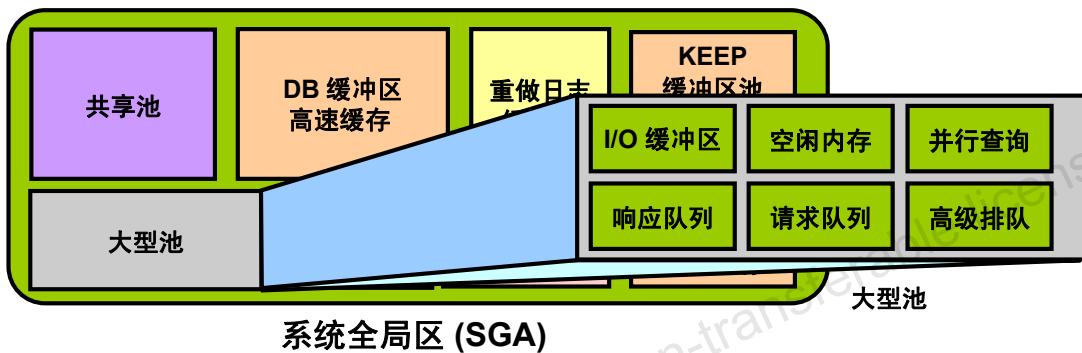
重做日志缓冲区是 SGA 中的一个循环缓冲区，用于存放有关对数据库所做更改的信息。此信息存储在重做条目中。重做条目包含重建（或重做）由 DML、DDL 或内部操作对数据库进行的更改所需的信息。需要时，将使用重做条目进行数据库恢复。

服务器进程更改缓冲区高速缓存时，将会生成重做条目，并将其写入 SGA 中的重做日志缓冲区。重做条目占用缓冲区中连续的顺序空间。日志写进程后台进程会将重做日志缓冲区写入磁盘上的活动重做日志文件（或文件组）中。

# 大型池

为以下对象提供大型内存分配:

- 共享服务器和 Oracle XA 接口的会话内存
- I/O 服务器进程
- Oracle DB 备份和还原操作



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 大型池

数据库管理员可以配置称为“大型池”的可选内存区，以便为以下对象提供大型内存分配:

- 共享服务器和 Oracle XA 接口（在事务处理与多个数据库交互时使用）的会话内存
- I/O 服务器进程
- Oracle DB 备份和还原操作
- 并行查询操作
- 高级排队内存表存储

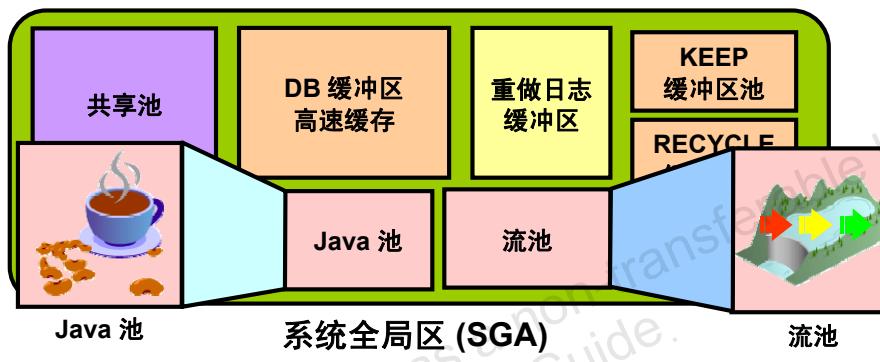
通过从大型池中为共享服务器、Oracle XA 或并行查询缓冲区分配会话内存，Oracle DB 可以主要使用共享池来缓存共享 SQL，并避免由于收缩共享 SQL 高速缓存而导致的性能开销。

此外，用于 Oracle DB 备份和还原操作、I/O 服务器进程和并行缓冲区的内存空间需要以数百 KB 的缓冲区进行分配。与共享池相比，大型池可以更好地满足此类大型内存请求。

大型池不是由最近最少使用 (LRU) 列表管理的。

## Java 池和流池

- Java 池内存用于存储 JVM 中所有特定于会话的 Java 代码和数据。
- Oracle Streams 以独占方式使用流池内存来执行以下操作：
  - 存储缓冲的队列消息
  - 为 Oracle Streams 进程提供内存



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

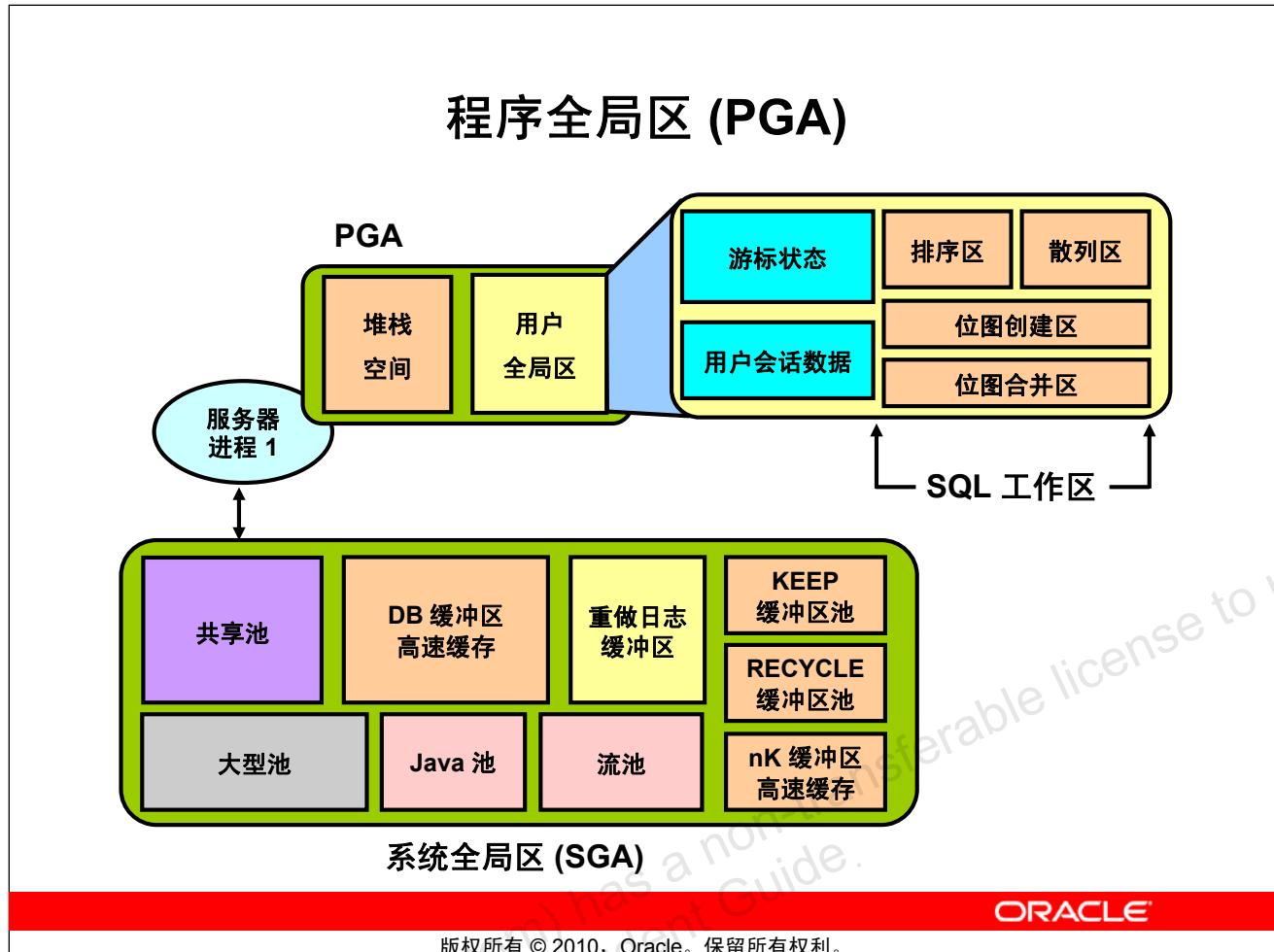
### Java 池和流池

Java 池内存用于存储 JVM 中所有特定于会话的 Java 代码和数据。Java 池内存的使用方式有多种，具体取决于 Oracle DB 的运行模式。

流池由 Oracle Streams 独占使用。流池存储缓冲的队列消息，并且为 Oracle Streams 捕获进程和应用进程提供内存。

除非对其进行专门配置，否则流池的大小从零开始。当使用 Oracle Streams 时，池大小会根据需要动态增长。

**注：**本课不会详细讨论 Java 编程和 Oracle Streams。



## 程序全局区 (PGA)

程序全局区 (PGA) 是一个专用内存区，其中包含服务器进程的数据及控制信息。每个服务器进程都有独立的 PGA。PGA 只能由相应的服务器进程访问，并且只有代表该服务器进程的 Oracle 代码可以读取它。开发人员的代码不能访问 PGA。

每个 PGA 都包含堆栈空间。在专用服务器环境中，连接到数据库实例的每个用户都有单独的服务器进程。对于这种类型的连接，PGA 包含一个名为用户全局区 (UGA) 的内存细分部分。UGA 包括以下部分：

- 游标区，用于存储游标的运行时信息
- 用户会话数据存储区，用于存储有关会话的控制信息
- SQL 工作区，用于处理 SQL 语句，其中包括：
  - 排序区，用于对数据排序的函数，如 ORDER BY 和 GROUP BY
  - 散列区，用于执行表的散列联接
  - 位图创建区，用于创建数据仓库常用的位图索引
  - 位图合并区，用于解析位图索引计划的执行

在共享服务器环境中，多个客户机用户共享服务器进程。在这种模式下，UGA 将移入 SGA（共享池，如果经过配置也可是大型池），PGA 仅包含堆栈空间。

## 小测验

包含服务器进程或后台进程的数据及控制信息的内存区域称为：

1. 共享池
2. PGA
3. 缓冲区高速缓存
4. 用户会话数据

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2

## 小测验

从数据文件读取到数据库缓冲区高速缓存中的是什么？

1. 行
2. 更改
3. 块
4. SQL

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 3

## 进程体系结构

- 用户进程
  - 是连接到 Oracle DB 的应用程序或工具
- 数据库进程
  - 服务器进程：连接到 Oracle 实例，在用户建立会话时启动
  - 后台进程：在启动 Oracle 实例时启动
- 守护程序/应用程序进程
  - 网络监听程序
  - Grid infrastructure 守护程序



版权所有 © 2010, Oracle。保留所有权利。

### 进程体系结构

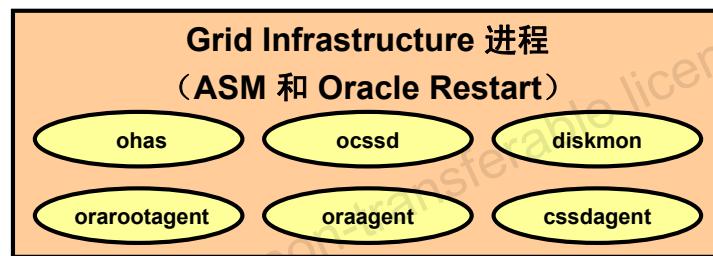
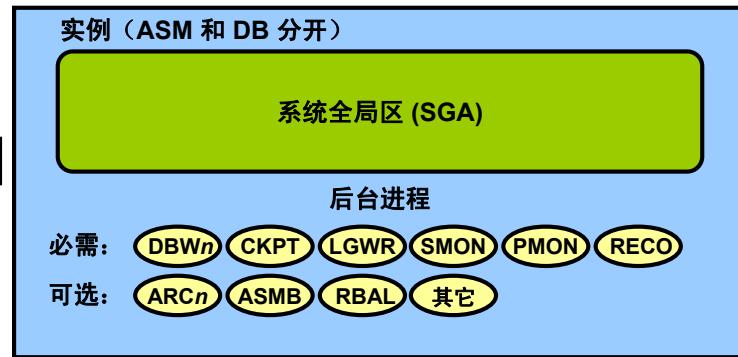
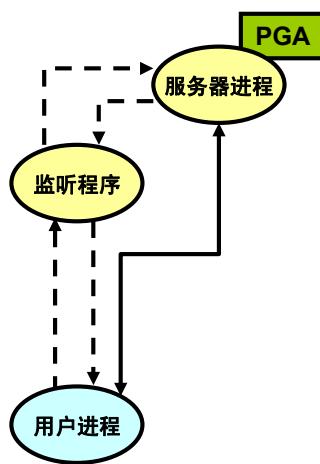
Oracle DB 系统中的进程可以主要分为三组：

- 运行应用程序或 Oracle 工具代码的用户进程。
- 运行 Oracle DB 服务器代码的 Oracle DB 进程（包括服务器进程和后台进程）。
- 非特定于单个数据库的 Oracle 守护程序和应用程序进程。

当用户运行应用程序或 Oracle 工具（如 SQL\*Plus）时，将使用术语“用户进程”来指代用户的应用程序。用户进程可以位于数据库服务器计算机上，也可以不位于该计算机上。Oracle DB 还会创建“服务器进程”以执行该用户进程发出的命令。此外，Oracle 服务器还为一个实例提供了一组“后台进程”。这些进程不仅彼此交互，还与操作系统交互，以便管理内存结构，异步执行 I/O 将数据写入磁盘，并执行其它需要完成的任务。对于不同的 Oracle DB 配置，进程结构也有所不同，具体取决于操作系统和选择的 Oracle DB 选件。已连接用户的代码可以配置为专用服务器或共享服务器。

- **专用服务器：**对于每个会话，运行数据库应用程序的用户进程都由执行 Oracle DB 服务器代码的专用服务器进程提供服务。
- **共享服务器：**不必为每个连接都提供一个专用服务器进程。分派程序会将多个传入网络会话请求定向到共享服务器进程池。共享服务器进程为所有客户机请求提供服务。

## 进程结构



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 进程结构

### 服务器进程

Oracle DB 创建服务器进程以处理连接到实例的用户进程的请求。用户进程代表连接到 Oracle DB 的应用程序或工具。它可以与 Oracle DB 在同一台计算机上，也可以在远程客户机上利用网络来访问 Oracle DB。用户进程首先与一个监听程序进程通信，在专用环境中该进程会创建一个服务器进程。

所创建的代表每个用户的应用程序的服务器进程可以执行以下一项或多项操作：

- 对通过应用程序发出的 SQL 语句进行语法分析并运行语句
- 从磁盘上的数据文件中将必要的数据块读取到 SGA 的共享数据库缓冲区中（如果这些数据块目前尚未在 SGA 中）
- 返回结果，使应用程序可以处理信息

### 后台进程

为了最大限度地提高性能并满足多个用户的需要，多进程 Oracle DB 系统使用一些称为“后台进程”的附加 Oracle DB 进程。一个 Oracle DB 实例可以有多个后台进程。

## 进程结构（续）

非 RAC、非 ASM 环境中的常见后台进程包括：

- 数据库写进程 (DBW $n$ )
- 日志写进程 (LGWR)
- 检查点进程 (CKPT)
- 系统监视器进程 (SMON)
- 进程监视器进程 (PMON)
- 恢复器进程 (RECO)
- 作业队列协调程序 (CJQ0)
- 作业从属进程 (Jnnn)
- 归档进程 (ARC $n$ )
- 队列监视器进程 (QM $n$ )

更高级的配置（如 RAC）中可能会有其它后台进程。有关后台进程的详细信息，请参见 V\$BGPRESS 视图。

有些后台进程是在启动实例时自动创建的，而另外一些则是根据需要创建的。

其它进程结构不是特定于单个数据库的，而是可以在同一个服务器上的多个数据库间共享的。Grid Infrastructure 进程和网络进程即属于此类。

Linux 和 UNIX 系统上的 Oracle Grid Infrastructure 进程包括：

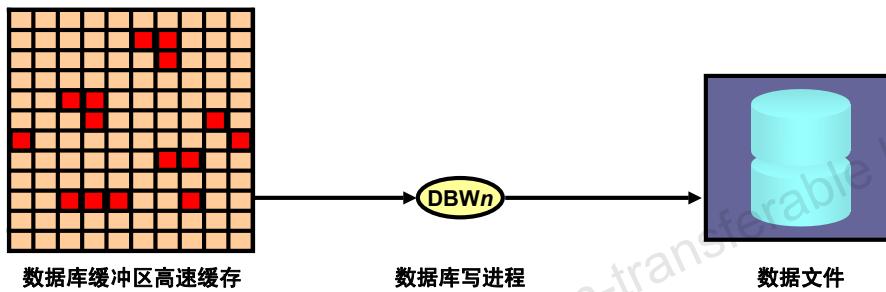
- ohasd: Oracle 高可用性服务守护程序，负责启动 Oracle Clusterware 进程
- ocssd: 集群同步服务守护程序
- diskmon: 磁盘监视守护程序，负责监视 HP Oracle Exadata Storage Server 的输入和输出
- cssdagent: 启动、停止和检查 CSS 守护程序 ocssd 的状态
- oraagent: 扩展集群件以支持 Oracle 特有的要求和复杂资源
- orarootagent: 一种专用的 Oracle 代理进程，可帮助管理 root 用户所拥有的资源（如网络）

注：有关后台进程的更详细列表，请参阅本课程的“Oracle 后台进程”附录，或者参阅《Oracle Database Reference》指南。

## 数据库写进程 (DBWn)

将数据库缓冲区高速缓存中经过修改的缓冲区（灰数据缓冲区）写入磁盘有两种方式：

- 在执行其它处理时异步执行
- 推进检查点



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据库写进程 (DBWn)

数据库写进程 (DBWn) 可以将缓冲区的内容写入数据文件。DBWn 进程负责将数据库缓冲区高速缓存中经过修改的缓冲区（灰数据缓冲区）写入磁盘。虽然对于大多数系统来说，一个数据库写进程 (DBW0) 已经足够；但是，如果系统需要频繁修改数据，也可以配置附加进程 (DBW1 到 DBW9 以及 DBWa 到 DBWz) 来改进写性能。这些附加 DBWn 进程在单处理器系统中没有用。

当数据库缓冲区高速缓存中的某个缓冲区被修改时，系统会将其标记为灰数据缓冲区，并将其添加到按 SCN 顺序排列的检查点队列的头中。因此，该顺序与这些更改的缓冲区的重做条目写入重做日志的顺序一致。当缓冲区高速缓存中的可用缓冲区的数量低于某个内部阈值（达到服务器进程认为很难获取可用缓冲区的程度）时，DBWn 会将不经常使用的缓冲区写入数据文件，写入顺序是从 LRU 列表的尾部开始，从而使进程可以在需要缓冲区时替换它们。DBWn 也会从检查点队列的尾部写入，以保护检查点向前推进。

SGA 中有一个内存结构保存了重做流中位置的重做字节地址 (RBA)，当实例发生故障时，将从该位置开始恢复。此结构充当指向重做的指针，并且由 CKPT 进程按照每三秒一次的频率写入控制文件。由于 DBWn 按照 SCN 顺序写入灰数据缓冲区，而重做按照 SCN 顺序执行，因此每当 DBWn 从 LRUW 列表写灰数据缓冲区时，还会将 SGA 内存结构中保持的指针前移，以便实例恢复（如果需要）从近似正确的位置开始读取重做，并避免不必要的 I/O。这称为“增量检查点”。

## 数据库写进程 (DBWn) (续)

注：还有其它一些 DBWn 可能执行写操作的情况（例如，当表空间被设置为只读或被置于脱机状态时）。在这些情况下，不会出现增量检查点，因为仅属于相应数据文件的灰数据缓冲区在写入数据库时的顺序与 SCN 顺序无关。

LRU 算法将更频繁访问的块保存在缓冲区高速缓存中，以尽可能地减少磁盘读取。可对表使用 CACHE 选项，以帮助延长块在内存中的保留时间。

DB\_WRITER\_PROCESSES 初始化参数指定了 DBWn 进程的数量。DBWn 进程的最大数量为 36。如果用户在启动过程中未指定该进程数，Oracle DB 将根据 CPU 和处理器组的数量来决定如何设置 DB\_WRITER\_PROCESSES。

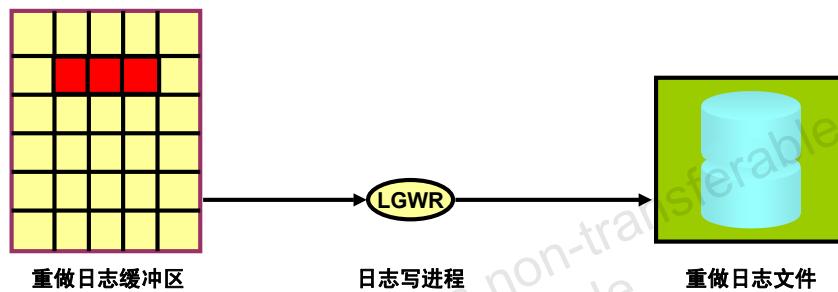
在以下情况下，DBWn 进程将灰数据缓冲区写入磁盘：

- 当服务器进程在扫描阈值数目的缓冲区之后找不到干净的可重用缓冲区时，会通知 DBWn 执行写操作。DBWn 在执行其它处理的同时，将灰数据缓冲区异步写入磁盘。
- DBWn 写缓冲区以推进检查点。检查点是重做线程（日志）中用于执行实例恢复的起始位置。该日志位置由缓冲区高速缓存中最旧的灰数据缓冲区确定。

在所有情况下，DBWn 均执行成批（多块）写操作以提高效率。多块写操作中写入的块数因操作系统而异。

## 日志写进程 (LGWR)

- 将重做日志缓冲区写入磁盘上的重做日志文件中
- 在以下情况下执行写操作:
  - 用户进程提交事务处理时
  - 重做日志缓冲区的三分之一已满时
  - DBWn 进程将经过修改的缓冲区写入磁盘之前
  - 每隔 3 秒



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 日志写进程 (LGWR)

日志写进程 (LGWR) 负责管理重做日志缓冲区，即将重做日志缓冲区条目写入磁盘上的重做日志文件。LGWR 会将上次写入后复制到缓冲区中的所有重做条目写入重做日志文件。

重做日志缓冲区是循环缓冲区。当 LGWR 将重做日志缓冲区中的重做条目写入重做日志文件时，服务器进程随后可以复制新条目，覆盖重做日志缓冲区中的那些已写入磁盘的条目。LGWR 的写入速度通常足够快，可以确保缓冲区中始终有空间可供新条目使用，即使对重做日志的访问量很大时也是如此。LGWR 将缓冲区的一个连续部分写入磁盘。

LGWR 在以下情况下执行写操作：

- 用户进程提交事务处理时
- 重做日志缓冲区的三分之一已满时
- DBWn 进程将经过修改的缓冲区写入磁盘（如果需要）之前
- 每隔 3 秒

## 日志写进程 (LGWR) (续)

在 DBW $n$  可以写入经过修改的缓冲区之前，必须先将与缓冲区更改相关联的所有重做记录写入磁盘（先行写协议）。如果 DBW $n$  发现一些重做记录尚未写入，则会通知 LGWR 将这些重做记录写入磁盘，并等待 LGWR 完成重做日志缓冲区的写入操作，然后才会写出数据缓冲区。LGWR 将向当前的日志组进行写入。如果该组中的某个文件已损坏或不可用，LGWR 将继续写入到该组中的其它文件，并在 LGWR 跟踪文件和系统预警日志中记录一个错误。如果某个组中的所有文件均已损坏，或者该组由于尚未归档而不可用，则 LGWR 无法继续工作。

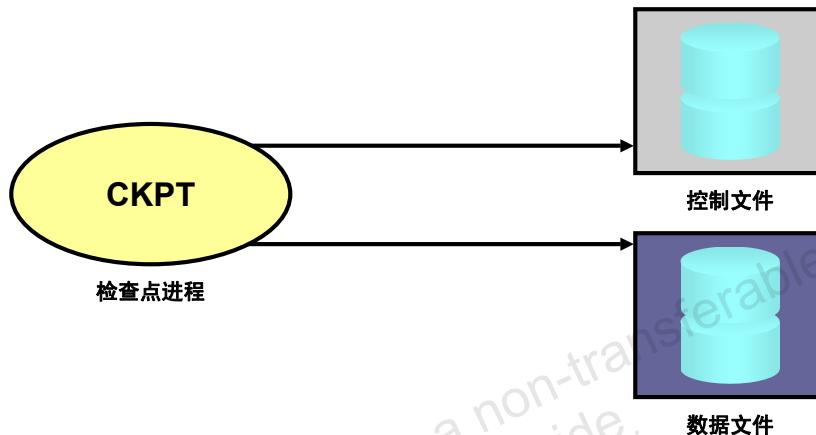
当用户发出 COMMIT 语句时，LGWR 会将一条提交记录放在重做日志缓冲区中，并立即将其记录随同事务处理的重做日志一起写入磁盘中。对数据块进行的相应更改将延迟，直到能够更为高效地写入这些更改时才会执行。这称为“快速提交机制”。包含事务处理提交记录的重做条目的原子写是单个事件，该事件可以确定事务处理是否已提交。Oracle DB 为提交的事务处理返回一个成功代码，尽管数据缓冲区尚未写入磁盘中。

如果需要更多缓冲区空间，LGWR 有时会在提交事务处理之前写入重做日志条目。仅当稍后提交了该事务处理之后，这些条目才会成为永久条目。当用户提交事务处理时，该事务处理将被分配一个系统更改号 (SCN)，Oracle DB 将该号码与事务处理的重做条目一起记录在重做日志中。SCN 记录在重做日志中，以便可以在 Real Application Cluster 和分布式数据库之间同步恢复操作。

当活动比较频繁时，LGWR 可以使用组提交来写入重做日志文件。例如，假设一个用户提交了一个事务处理。LGWR 必须将该事务处理的重做条目写入磁盘。但在此时，又有其他用户发出了 COMMIT 语句。但是，LGWR 在完成其上一个写操作之前，无法将这些事务处理的重做条目写入重做日志文件以提交这些事务处理。在第一个事务处理的条目写入重做日志文件之后，正在等待（尚未提交）的事务处理的整个重做条目列表将通过一次操作写入磁盘，这比分别处理各个事务处理条目所需的 I/O 要少。因此，Oracle DB 可以将磁盘 I/O 降至最少，并最大限度地提高 LGWR 的性能。如果提交请求的速率一直较高，则从重做日志缓冲区（由 LGWR）进行的每个写操作都可能包含多个提交记录。

## 检查点进程 (CKPT)

- 检查点信息记录在以下位置
  - 控制文件
  - 每个数据文件头



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 检查点进程 (CKPT)

“检查点”是一种数据结构，它定义了数据库的重做线程中的系统更改号 (SCN)。检查点记录在控制文件和每个数据文件头中。它们是恢复操作的关键元素。

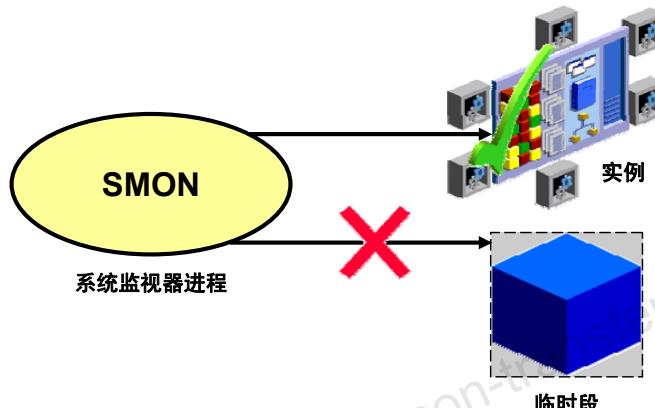
遇到检查点时，Oracle DB 必须更新所有数据文件的头，以记录该检查点的详细信息。

这是由 CKPT 进程完成的。CKPT 进程不会将块写入磁盘；该工作始终由 DBWn 执行。

文件头中记录的 SCN 可保证在该 SCN 之前对数据库块进行的所有更改都已写入到磁盘中。

## 系统监视器进程 (SMON)

- 在实例启动时执行恢复
- 清除不使用的临时段



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

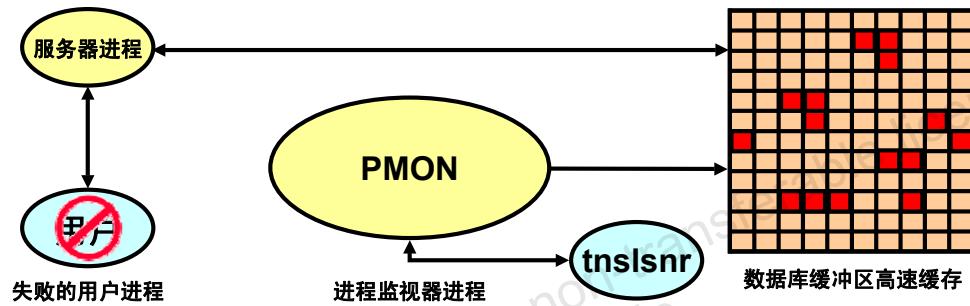
### 系统监视器进程 (SMON)

系统监视器进程 (SMON) 可在实例启动时执行恢复（如果需要）。SMON 还负责清除不再使用的临时段。如果在实例恢复过程中由于文件读取错误或脱机错误而跳过了任何已终止的事务处理，则 SMON 将在表空间或文件重新联机时恢复这些事务处理。

SMON 会定期检查，以查看是否需要该进程。其它进程在检测到需要 SMON 时也可以调用它。

## 进程监视器进程 (PMON)

- 在用户进程失败时执行进程恢复
  - 清除数据库缓冲区高速缓存
  - 释放该用户进程使用的资源
- 监视会话，查看是否发生空闲会话超时
- 在监听程序中动态注册数据库服务



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 进程监视器进程 (PMON)

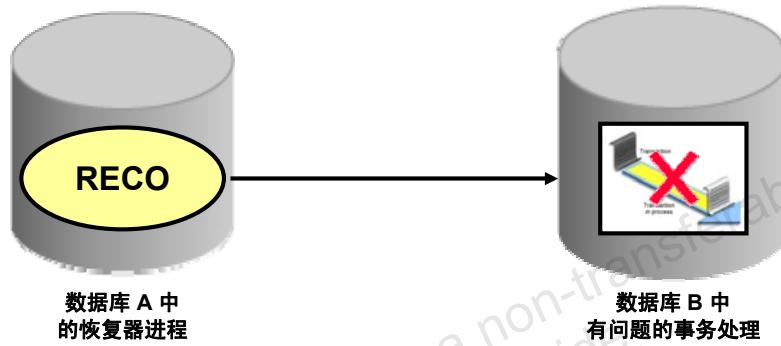
进程监视器进程 (PMON) 在用户进程失败时执行进程恢复。PMON 负责清除数据库缓冲区高速缓存和释放该用户进程占用的资源。例如，PMON 会重置活动事务处理表的状态，释放锁，并从活动进程列表中删除该进程 ID。

PMON 定期检查分派程序和服务器进程的状态，并重新启动任何已停止运行（除了 Oracle DB 故意终止）的分派程序和服务器进程。PMON 还会在网络监听程序中注册有关实例和分派程序进程的信息。

与 SMON 一样，PMON 定期检查以查看是否需要运行；如果其它进程检测到需要该进程，也可以调用它。

## 恢复器进程

- 用于分布式数据库配置
- 自动连接到有问题的分布式事务处理中涉及的其它数据库
- 自动解决所有有问题的事务处理
- 删除对应于有问题的事务处理的所有行



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

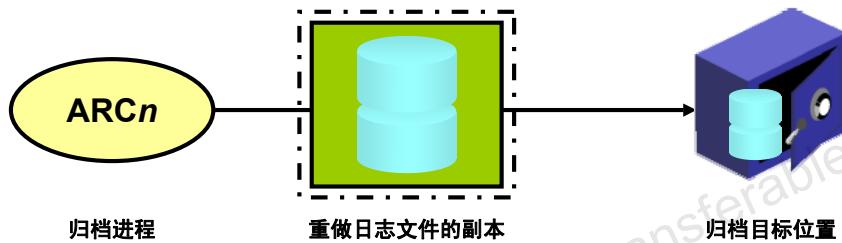
### 恢复器进程 (RECO)

恢复器进程 (RECO) 是一个用于分布式数据库配置的后台进程，它可以自动解决涉及分布式事务处理的故障。实例的 RECO 进程会自动连接到有问题的分布式事务处理中涉及的其它数据库。当 RECO 进程在涉及到的数据库服务器之间重新建立连接后，它会自动解决所有有问题的事务处理，并从每个数据库的暂挂事务处理表中删除所有对应于已解决的有问题事务处理的行。

如果 RECO 进程无法与远程服务器连接，RECO 会在某个计时间隔之后自动尝试重新连接。但是，RECO 在再次尝试另一个连接之前，会等待一段时间，该时间会随尝试次数不断增加（呈幂指数增长）。

## 归档进程 (ARCn)

- 在发生日志切换之后，将重做日志文件复制到指定的存储设备
- 可以收集事务处理重做数据，并将该数据传输到备用目标位置



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

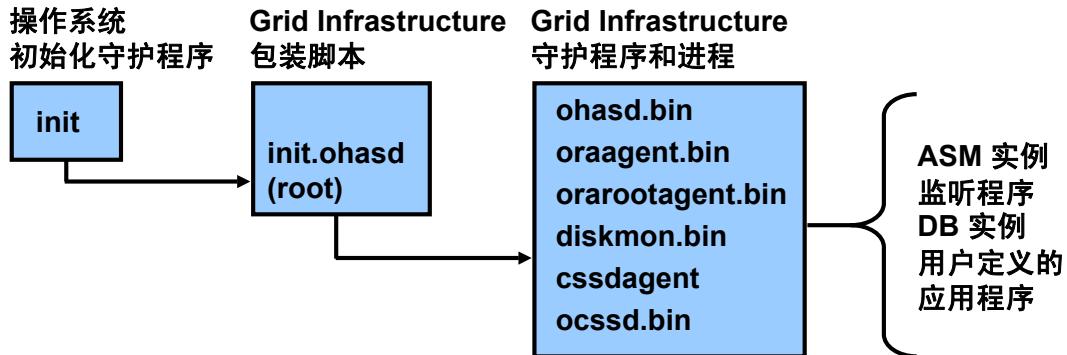
### 归档进程 (ARCn)

发生日志切换之后，归档进程 (ARCn) 会将重做日志文件复制到指定的存储设备。仅当数据库处于 ARCHIVELOG 模式且已启用自动归档时，才会存在 ARCn 进程。

如果您预计归档的工作负荷很重（例如在成批加载数据期间），可以增加最大归档进程数。此外，也可以有多个归档日志目标位置。建议每个目标位置至少有一个归档进程。默认设置是四个归档进程。

## 进程启动顺序

- Oracle Grid Infrastructure 由 OS 初始化守护程序启动。



- Oracle Grid Infrastructure 安装程序会修改 /etc/inittab 文件，以确保每次启动计算机时在相应运行级别启动 Oracle Grid Infrastructure。

```
# cat /etc/inittab
...
h1:35:respawn:/etc/init.d/init.ohasd run >/dev/null 2>&1 </dev/null
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 进程启动顺序

安装 Oracle Grid Infrastructure 时，会在操作系统的 /etc/inittab 文件中放入条目，以启动包装脚本。该包装脚本负责设置环境变量，然后启动 Oracle Grid Infrastructure 守护程序和进程。

使用命令停止 Oracle Grid Infrastructure 时，守护程序会停止，但包装脚本进程仍会运行。

UNIX 下的 /etc/inittab 文件格式如下：

```
id : run levels : action : process with parameters
```

由于包装脚本是使用 respawn 操作启动的，因此终止后，它都会重新启动。

在 Oracle Grid Infrastructure 守护程序启动以后，有些将在 root 用户身份下以实时优先级运行，而另一些将在 Oracle Grid Infrastructure 所有者身份下以用户模式优先级运行。

在 Windows 平台上，使用的是操作系统服务而非初始化包装脚本，而且守护程序是二进制的可执行文件。

注：不支持直接执行包装脚本。

## 数据库存储体系结构



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据库存储体系结构

构成 Oracle DB 的文件可划分为以下类别：

- 控制文件**：包含与数据库本身相关的数据，即物理数据库结构信息。这些文件对数据库至关重要。没有这些文件，就无法打开数据文件来访问数据库中的数据。控制文件还可以包含与备份相关的元数据。
- 数据文件**：包含数据库的用户或应用程序数据，以及元数据和数据字典。
- 联机重做日志文件**：用于进行数据库的实例恢复。如果数据库服务器发生崩溃，但未丢失任何数据文件，实例便可使用这些文件中的信息恢复数据库。

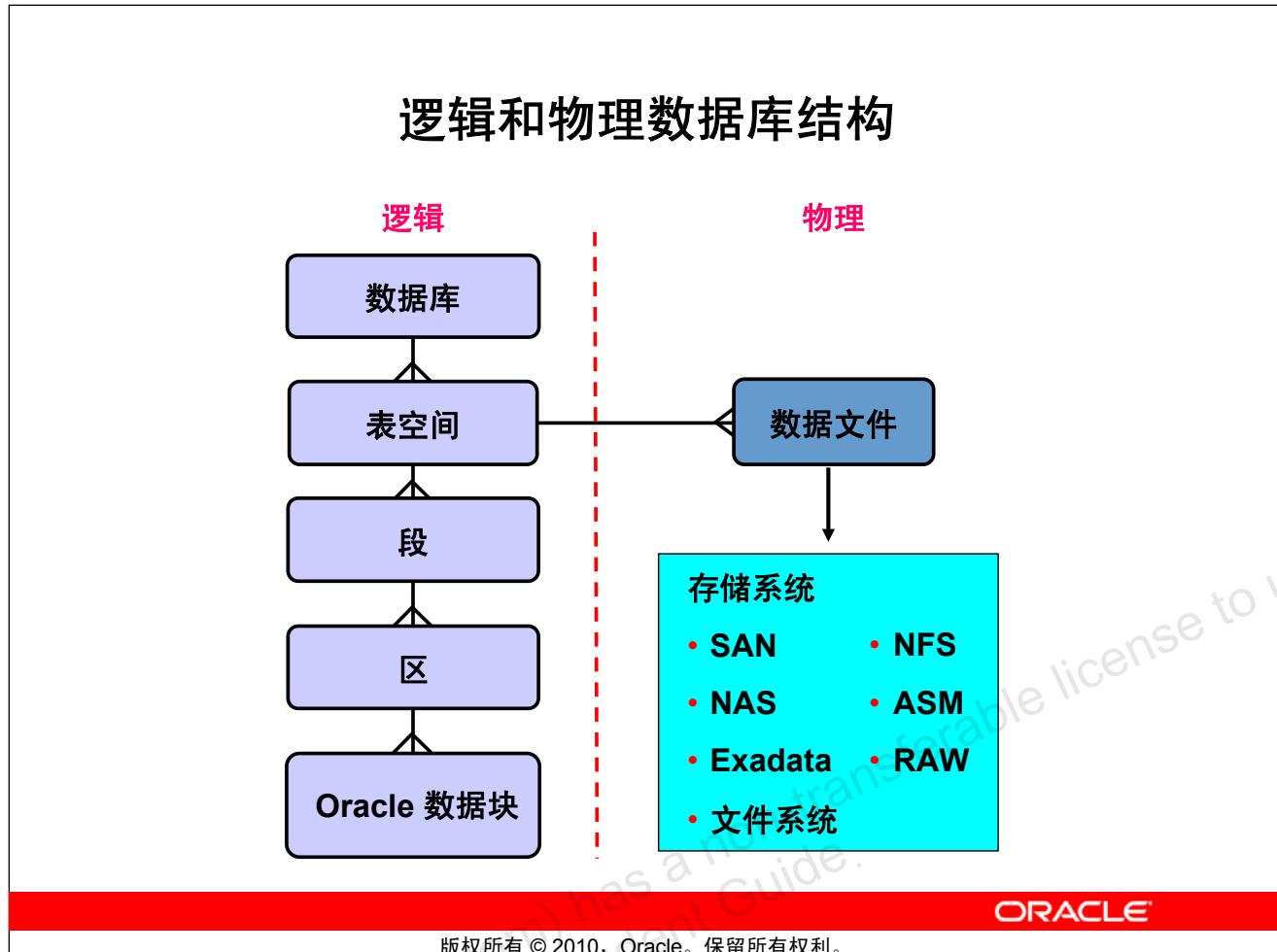
下列附加文件对成功运行数据库非常重要：

- 参数文件**：用于定义实例启动时的配置。
- 口令文件**：允许用户使用 sysdba、sysoper 和 sysasm 角色远程连接到实例并执行管理任务。
- 备份文件**：用于进行数据库恢复。介质出现故障或因用户错误而损坏或删除原始文件时，通常会还原备份文件。
- 归档重做日志文件**：包含实例发生的数据更改（重做）的实时历史记录。使用这些文件和数据库备份，可以恢复丢失的数据文件。也就是说，使用归档日志可以恢复还原的数据文件。

## 数据库存储体系结构（续）

- **跟踪文件：**每个服务器和后台进程都可以写入一个关联的跟踪文件。当进程检测到内部错误时，会将有关该错误的信息转储到相应的跟踪文件中。写入跟踪文件的一些信息是为数据库管理员提供的，而其它信息是为 Oracle 支持服务提供的。
- **预警日志文件：**这些文件包含特殊的跟踪条目。数据库的预警日志是一个按时间顺序记录消息和错误的日志。Oracle 建议您定期检查此预警日志。

注：参数、口令、预警和跟踪文件将在其它课中讨论。



## 逻辑和物理数据库结构

数据库具有逻辑结构和物理结构。

### 数据库、表空间和数据文件

本幻灯片对数据库、表空间和数据文件之间的关系进行了说明。每个数据库都在逻辑上分为两个或多个表空间。在每个表空间均显式创建一个或多个数据文件，以在物理上存储表空间中所有逻辑结构的数据。对于 TEMPORARY 表空间，不创建数据文件，而是创建临时文件。表空间的数据文件可以采用任何受支持的存储技术进行物理存储。

### 表空间

数据库分为多个逻辑存储单元，这些单元称为“表空间”，用于对相关逻辑结构或数据文件进行分组。例如，表空间一般会将一个应用程序的所有段分成一组，以简化一些管理操作。

### 数据块

Oracle DB 的数据存储在“数据块”中，数据块是粒度最低的一级。一个数据块对应于磁盘中特定字节数的物理空间。每个表空间的数据块大小是在表空间创建时指定的。数据库以 Oracle 数据块为单位使用和分配空闲数据库空间。

## 逻辑和物理数据库结构（续）

### 区

逻辑数据库空间的下一级是“区”。区是特定数量的相邻 Oracle 数据块（通过一次分配获得），用于存储特定类型的信息。区中的 Oracle 数据块在逻辑上是相邻的，但在物理上可以分布在磁盘上的不同位置（RAID 条带化和文件系统实施会导致此现象）。

### 段

逻辑数据库存储中区的上一级称为“段”。一个段是为某个逻辑结构分配的一组区。例如：

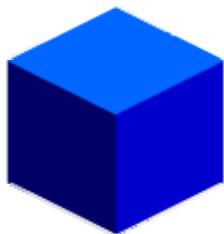
- **数据段：**每个非集群的、不按索引组织的表都有一个数据段，但外部表、全局临时表和分区表除外，这些表中的每个表都有一个或多个段。表中的所有数据都存储在相应数据段的区中。对于分区表，每个分区都有一个数据段。每个集群也都有一个数据段。集群中每个表的数据都存储在集群的数据段中。
- **索引段：**每个索引都有一个索引段，存储其所有数据。对于分区索引，每个分区都有一个索引段。
- **还原段：**系统会为每个数据库实例创建一个 UNDO 表空间。该表空间包含大量用于临时存储还原信息的还原段。还原段中的信息用于生成读一致性数据库信息，以便在数据库恢复过程中回退用户未提交的事务处理。
- **临时段：**临时段是 SQL 语句需要临时工作区来完成执行时由 Oracle DB 创建的。语句完成执行后，临时段的区将返回到实例以备将来使用。您可以为每个用户指定一个默认临时表空间，或指定一个在数据库范围内使用的默认临时表空间。

**注：**另外还有一些上面未列出的其它类型的段。此外，还有一些方案对象，如视图、程序包和触发器等，虽然它们是数据库对象，但不被视为段。段拥有单独的磁盘空间分配。其它对象则以行的形式存储在系统元数据段中。

Oracle DB 服务器对空间进行动态分配。如果段中的现有区已满，则会再增加一些区。因为区是根据需要来分配的，因此段中的区在磁盘上可能是相邻的，也可能是不相邻的，它们可以来自属于同一个表空间的不同数据文件。

## 段、区和块

- 段存在于表空间中。
- 段是区的集合。
- 区是数据块的集合。
- 数据块映射到磁盘块。



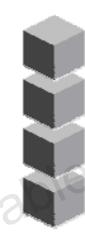
段



区



数据块

磁盘块  
(文件系统存储)

版权所有 © 2010, Oracle。保留所有权利。

### 段、区和块

一部分数据库对象（例如表和索引）以段的形式存储在表空间中。每个段都包含一个或多个区。区由相邻的数据块组成，这意味着每个区只能存在于一个数据文件中。数据块是数据库中最小的 I/O 单元。

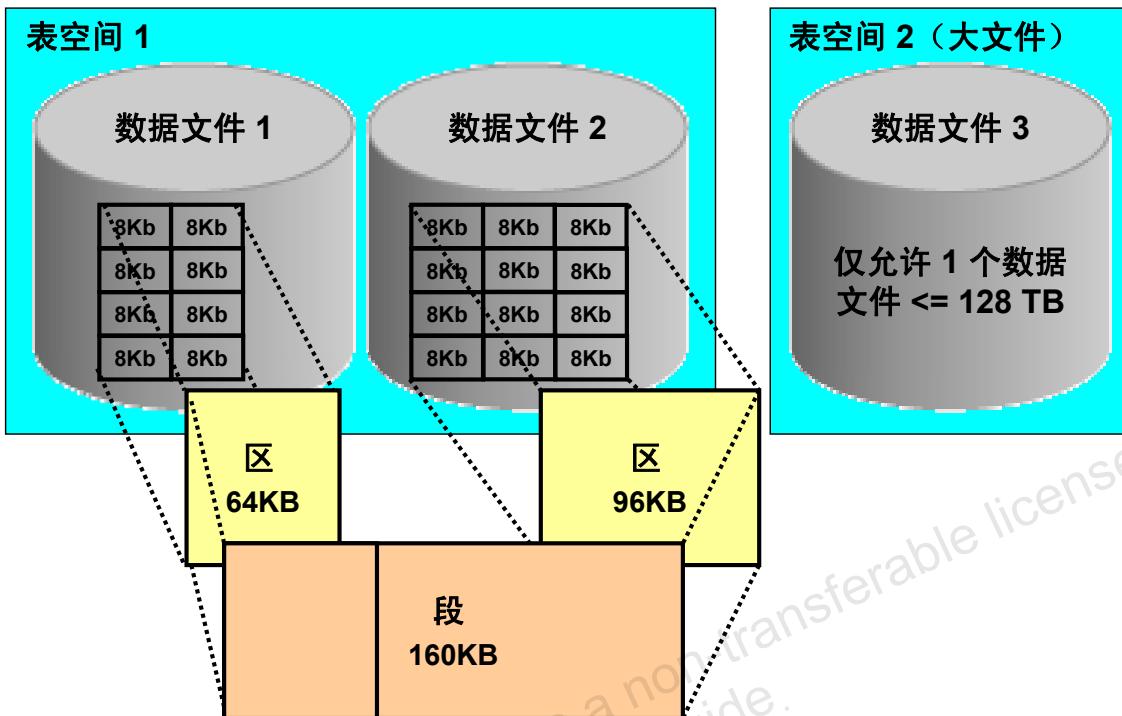
数据库从操作系统 (OS) 请求数据块集时，OS 会将数据块集映射到存储设备上的实际文件系统或磁盘块。因此，您不需要知道数据库中任何数据的物理地址。这还意味着一个数据文件可以在多个磁盘上进行条带化或创建镜像。

数据块的大小可以在创建数据库时设置。对大多数数据库而言，默认的 8 KB 块大小是足够的。如果数据库用于支持拥有大型表和索引的数据仓库应用程序，则最好设置较大的块大小。

如果数据库用于支持执行随机读写的事务处理应用程序，则最好指定较小的块大小。块大小的最大值取决于您的操作系统。Oracle 块大小的最小值为 2 KB，这个值很少使用。

您可以设置非标准块大小的表空间。有关详细信息，请参阅《Oracle 数据库管理员指南》。

## 表空间和数据文件



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 表空间和数据文件

数据库被划分为多个“表空间”，表空间是可用于将相关逻辑结构组合在一起的逻辑存储单元。每个数据库都在逻辑上分为两个或多个表空间：SYSTEM 和 SYSAUX 表空间。在每个表空间均显式创建一个或多个数据文件，以在物理上存储表空间中所有逻辑结构的数据。

幻灯片中的图形说明了由两个数据文件组成的表空间 1。一个大小为 160 KB 的段跨越两个数据文件，由两个区组成。第一个区位于第一个数据文件中，大小为 64 KB；第二个区位于第二个数据文件中，大小为 96 KB。两个区都由若干相邻的 8Kb Oracle 块组成。

**注：**您还可以创建大文件表空间，这种表空间只有一个通常非常大的文件。该文件的大小可达到行 ID 体系结构允许的最大大小。此最大大小是表空间的块大小乘以  $2^{36}$ ，如果块大小为 32 KB，则最大大小为 128 TB。传统的小文件表空间（默认值）可以包含多个数据文件，但这些文件都不大。有关大文件表空间的详细信息，请参阅《Oracle 数据库管理员指南》。

## SYSTEM 和 SYSAUX 表空间

- SYSTEM 和 SYSAUX 表空间是在创建数据库时创建的必需存在的表空间。这些表空间必须联机。
- SYSTEM 表空间用于核心功能（例如数据字典表）。
- 辅助的 SYSAUX 表空间用于附加的数据库组件（如 Oracle Enterprise Manager Repository）。
- 不建议使用 SYSTEM 和 SYSAUX 表空间来存储应用程序的数据。



版权所有 © 2010, Oracle。保留所有权利。

### SYSTEM 和 SYSAUX 表空间

每个 Oracle DB 都必须包含一个 SYSTEM 表空间和一个 SYSAUX 表空间。这两个表空间是在创建数据库时自动创建的。系统默认设置是创建小文件表空间。您也可以创建大文件表空间，这样可通过 Oracle DB 管理超大文件。

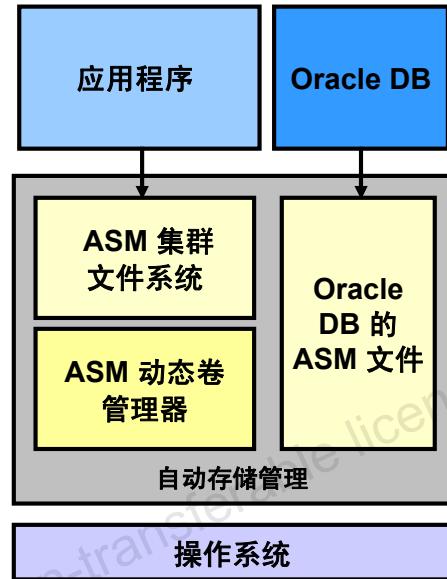
表空间可以处于联机（可访问）状态，也可以处于脱机（不可访问）状态。打开数据库时，SYSTEM 表空间始终处于联机状态。这个表空间存储了支持数据库核心功能的表，如数据字典表。

SYSAUX 表空间是 SYSTEM 表空间的辅助表空间。SYSAUX 表空间存储了许多数据库组件，要使所有数据库组件正常运行，该表空间必须处于联机状态。不建议使用 SYSTEM 和 SYSAUX 表空间来存储应用程序的数据。要存储应用程序的数据，可以另外创建表空间。

**注：** SYSAUX 表空间可以脱机以执行表空间恢复，而 SYSTEM 表空间则不能。这两种表空间都不能设置为只读。

## 自动存储管理

- 是可移植的高性能集群文件系统
- 管理 Oracle DB 文件
- 通过 ASM 集群文件系统 (ACFS) 管理应用程序文件
- 将数据分布到各个磁盘中以平衡负载
- 建立数据镜像以防范故障
- 解决存储管理挑战



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 自动存储管理

自动存储管理 (ASM) 为 Oracle DB 文件提供文件系统与卷管理器纵向集成。ASM 可管理单个对称多处理 (SMP) 计算机，或管理集群的多个节点来支持 Oracle Real Application Clusters (RAC)。

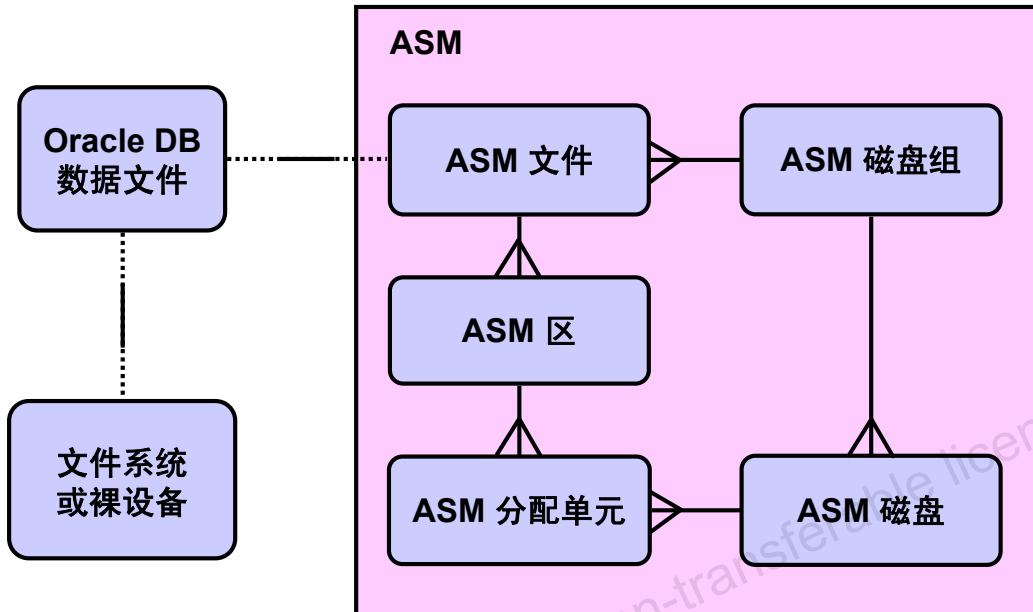
Oracle ASM 集群文件系统 (ACFS) 是一种多平台、可伸缩的文件系统和存储管理技术，该技术扩展了 ASM 的功能，可支持 Oracle DB 外部的应用程序文件，如可执行文件、报表、BFILE、视频、音频、文本、图像以及其它一般用途的应用程序文件数据。

ASM 在所有可用资源中分布输入/输出 (I/O) 负载，既免除了手动优化 I/O 又优化了性能。ASM 帮助 DBA 管理动态数据库环境，让 DBA 在不关闭数据库的情况下，通过增加数据库的大小来调整存储分配。

ASM 可以维护数据的冗余副本提供容错能力，也可以构建在供应商提供的存储机制之上。数据管理是通过为各类数据选择所需的可靠性和性能指标来实现的，而不是逐个文件地进行人工交互。

通过使手动完成的存储工作自动化，ASM 功能节省了 DBA 的时间，从而提高了管理员的能力，使其可以管理更多和更大的数据库，而且效率也更高。

## ASM 存储组件



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

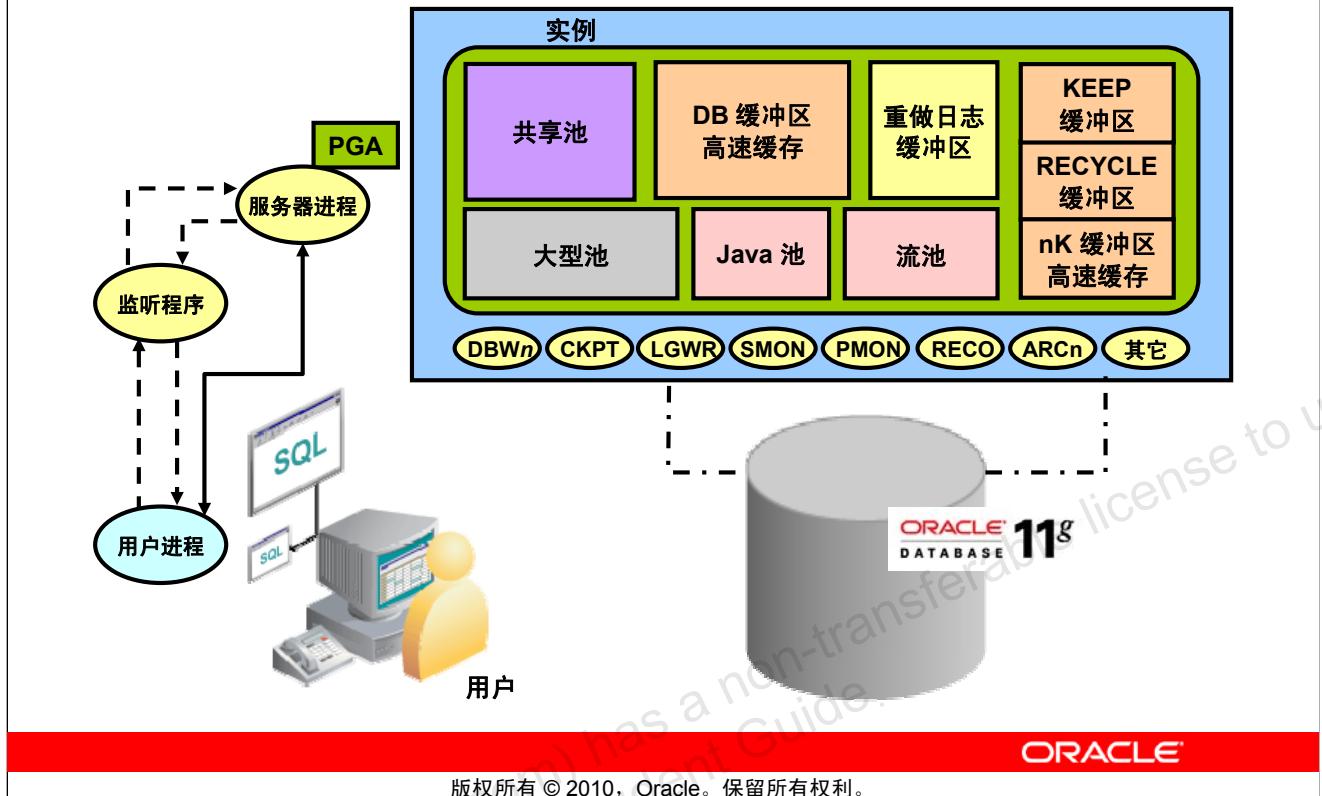
### ASM 存储组件

ASM 不会妨碍任何现有的数据库功能。现有数据库能够像平常一样工作。新文件可以被创建为 ASM 文件，而现有文件既可以按原有方式进行管理，也可以移植至 ASM。

上图说明了 Oracle DB 数据文件与 ASM 存储组件之间的关系。鸦脚标记代表一对多关系。Oracle DB 数据文件与存储在操作系统的文件系统中的文件或 ASM 文件之间是一对一关系。

Oracle ASM 磁盘组是作为一个逻辑单元进行管理的一个或多个 Oracle ASM 磁盘的集合。磁盘组中的数据结构是自包含的，使用部分空间来满足元数据需求。Oracle ASM 磁盘是为 Oracle ASM 磁盘组预配的存储设备，可以是物理磁盘，也可以是分区、存储阵列中的逻辑单元号 (LUN)、逻辑卷 (LV) 或连接到网络的文件。每个 ASM 磁盘被分成许多 ASM 分配单元 (AU)，该单元是 ASM 可以分配的最小相邻磁盘空间量。在创建 ASM 磁盘组时，可以将 ASM 分配单元的大小设置为 1、2、4、8、16、32 或 64 MB，具体取决于磁盘组的兼容级别。一个或多个 ASM 分配单元即形成一个 ASM 区。Oracle ASM 区是用于存放 Oracle ASM 文件内容的裸存储。Oracle ASM 文件由一个或多个文件区组成。为了支持非常大的 ASM 文件，可以使用可变大小区，区大小可等于 AU 大小的 1 倍、4 倍和 16 倍。

## 与 Oracle DB 交互：内存、进程和存储



### 与 Oracle DB 交互

以下示例在最基本的层面上描述了 Oracle DB 的操作。在该示例展示的 Oracle DB 配置中，用户和关联服务器进程在不同计算机上运行，这些计算机通过网络连接。

1. 在安装了 Oracle DB 的节点（通常称为“主机”或“数据库服务器”）上启动了一个实例。
2. 用户启动一个应用程序，从而衍生了一个用户进程。该应用程序尝试与服务器建立一个连接。（此连接可以是本地连接、客户机/服务器连接或来自中间层的三层连接）。
3. 服务器运行一个具有相应 Oracle Net 服务处理程序的监听程序。监听程序检测到应用程序发出的连接请求，并创建一个代表用户进程的专用服务器进程。
4. 用户运行一条 DML 类型的 SQL 语句并提交事务处理。例如，用户更改表中的客户地址并提交更改。
5. 服务器进程接收该语句，并检查共享池（一个 SGA 组件）中是否有包含相同 SQL 语句的共享 SQL 区域。如果找到共享 SQL 区域，服务器进程将检查用户对于所请求数据的访问权限，然后使用现有的共享 SQL 区域处理该语句。如果未找到共享 SQL 区域，则为该语句分配一个新的共享 SQL 区域，以便对该语句进行语法分析和处理。
6. 服务器进程从实际数据文件（表）或数据库缓冲区高速缓存中存储的值中检索任何必需的数据值。

## 与 Oracle DB 交互（续）

7. 服务器进程修改 SGA 中的数据。因为已提交事务处理，所以日志写进程 (LGWR) 会立即在重做日志文件中记录该事务处理。数据库写进程 (DBWn) 在一个高效的时机将修改后的块永久写入磁盘。
8. 如果事务处理成功，服务器进程将通过网络向应用程序发送一条消息。如果事务处理不成功，则传送一条错误消息。
9. 在整个过程中，其它后台进程也在运行，监视是否有需要干预的情况。此外，数据库服务器管理其他用户的事务处理，并防止请求相同数据的事务处理之间发生争用。

## 小测验

进程监视器进程 (PMON):

1. 在实例启动时执行恢复
2. 在用户进程失败时执行进程恢复
3. 自动解决所有有问题的事务处理
4. 将重做日志缓冲区写入重做日志文件中

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小测验

哪些类型的实例可以访问 ASM 文件？

1. 只有 RDBMS 实例
2. 只有 ASM 实例
3. RDBMS 实例和 ASM 实例都可以

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 3

## 小结

在本课中，您应该已经学会：

- 列出 Oracle DB 的主要体系结构组件
- 说明内存结构
- 描述后台进程
- 将逻辑存储结构与物理存储结构关联起来
- 描述 ASM 存储组件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 1：概览

这是一个书面练习，含有关于以下内容的问题：

- 数据库体系结构
- 内存
- 进程
- 文件结构

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



## 安装 Oracle 软件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 描述数据库管理员 (DBA) 的角色，说明典型的任务和工具
- 计划 Oracle 软件安装
- 安装用于独立服务器的 Oracle Grid Infrastructure
- 安装 Oracle DB 软件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## Oracle DB 管理员的任务

设计、实施和维护 Oracle DB 的方法包括以下任务：

1. 评估数据库服务器硬件
2. 安装 Oracle 软件
3. 制定数据库和安全策略计划
4. 创建、移植和打开数据库
5. 备份数据库
6. 登记系统用户和制定用户访问 Oracle Network 的计划
7. 实施数据库设计
8. 从数据库故障中进行恢复
9. 监视数据库性能



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle DB 管理员的任务

DBA 一般负责安装 Oracle 软件和创建数据库。作为一个 DBA，可能要负责创建数据库存储结构，如表空间。此外，还可能要负责创建用于保存应用程序数据的方案或对象集。

您必须确保用户均可以使用数据库。可以通过启动数据库、定期备份数据库和监视数据库性能达到此目的。这些任务应该在安全策略框架内执行。

在本课程每一课的学习过程中，您将分别学习如何执行上述每一项任务。有关本幻灯片中列出的每项任务的其它信息，请参阅《Oracle 数据库管理员指南》。

本课将重点介绍如何进行安装。为了完成此核心任务，请先考虑完成以下子任务：

- 了解安装应如何适应组织的总体技术体系结构
- 复查（并更新）容量计划
- 选择数据库软件（所需的版本和选件）
- 确保满足所有选定元素的系统要求

## 用于管理 Oracle DB 的工具

- Oracle Universal Installer
- Database Configuration Assistant
- Database Upgrade Assistant
- Oracle Net Manager
- Oracle Net Configuration Assistant
- Oracle Enterprise Manager
- Server Control 实用程序
- SQL\*Plus
- Recovery Manager
- 数据泵
- SQL\*Loader



版权所有 © 2010, Oracle。保留所有权利。

## 用于管理 Oracle DB 的工具

您可以使用以下工具进行安装和升级：

- **Oracle Universal Installer (OUI):** 安装 Oracle 软件和选件；可以自动启动 Database Configuration Assistant 来创建数据库
- **Database Configuration Assistant (DBCA):** 通过 Oracle 提供的模板创建数据库，使您能够复制预配置的种子数据库（或者也可以创建自己的数据库和模板。）
- **Database Upgrade Assistant (DBUA):** 引导您将现有数据库升级至 Oracle 新版本
- **Oracle Net Manager (netmgr):** 配置 Oracle DB 与应用程序的网络连接
- **Oracle Net Configuration Assistant (NetCA):** 一个基于向导的图形化工具，用于配置和管理 Oracle Network 配置

## 用于管理 Oracle DB 的工具（续）

以下工具用于管理 Oracle 实例和数据库：

- **Oracle Enterprise Manager (EM):** 将图形控制台、代理、公用服务和工具组合在一起，为管理 Oracle 产品提供了一个集成的综合性系统管理平台。在安装 Oracle 软件、创建或升级数据库以及配置网络之后，可以将 EM 用作管理数据库的单一界面。除了为执行 SQL 命令提供基于 Web 的用户界面外，它还能与其它用于管理数据库的 Oracle 组件（例如，Oracle Recovery Manager 和 Oracle Scheduler）进行交互。
- 用于管理 Oracle DB 的主要 EM 工具包括：
  - **Enterprise Manager Database Console:** 用于管理一个数据库
  - **Enterprise Manager Grid Control:** 用于同时管理多个数据库
- **Server Control 实用程序 (srvctl):** 标准的命令行界面，可用于启动和停止数据库和实例、管理 ASM 实例、管理配置信息以及移动或删除实例和服务。您也可以使用 SRVCTL 来添加服务和管理配置信息
- **SQL\*Plus:** 用于管理数据库的标准命令行界面
- **Oracle Recovery Manager (RMAN):** 一款可为以下任务提供完整解决方案的 Oracle 工具：对整个数据库或特定的数据库文件进行备份、还原和恢复
- **数据泵:** 用于在数据库之间进行高速数据传输（例如，您可能需要导出某个表，然后将其导入另一数据库。）
- **SQL\*Loader:** 用于将来自外部文件的数据加载到 Oracle DB 中；它是可用于将数据加载到数据库表中的多个 Oracle 实用程序之一
- **命令行工具:**
  - 要管理 Enterprise Manager，请使用：  
`emctl start | status | stop dbconsole`
  - 要管理监听程序，请使用：  
`lsnrctl start | status | stop`

## 制定安装计划

- 要安装什么 Oracle 软件？
- 涉及的硬件是否能达到规范的最低要求？
- 在安装多个产品时，是否有推荐的安装顺序？
- 是否存在必须由 DBA 以外的其他人执行的先决条件步骤？



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 制定安装计划

在开始安装 Oracle 软件之前，应向自己提出以下问题来帮助制定安装计划：

- 要安装什么 Oracle 软件？

在本课程中，假定您要安装 Oracle DB。Oracle 推荐的最佳做法是使用自动存储管理 (ASM) 作为存储技术。这需要使用集群软件介质安装 Oracle Grid Infrastructure。此项安装将安装 ASM 所需的组件以及 Oracle Restart。

- 涉及的硬件是否能达到规范的最低要求？

确定安装过程中涉及的所有硬件，并确保达到规范建议的最低要求。

- 在安装多个产品时，是否有推荐的安装顺序？

只要可行，建议先安装 Oracle Grid Infrastructure，然后再安装 Oracle DB 软件。按此顺序安装意味着新创建的数据库可以配置为使用 ASM 硬盘组，并且此数据库会在 Oracle Restart 中自动注册。如果先安装 Oracle DB，再安装 Oracle Grid Infrastructure，则需要执行手动配置步骤在 Oracle Restart 中注册数据库。如果希望此现有数据库使用 ASM 磁盘组进行存储，则需要执行移植步骤。

- 是否存在必须由 DBA 以外的其他人执行的先决条件步骤？

按照推荐的职责分离准则，DBA 可能不会负责配置安装所要用的硬件和存储设备。在安装 Oracle Grid Infrastructure 之前，需要存储管理员执行一些必需的配置步骤来配置所需的磁盘分区。有关详细信息，请参阅《Oracle 数据库安装指南》。

## Oracle Grid Infrastructure 和 Oracle DB 安装：系统要求

- 内存要求：
  - 使用 Oracle Enterprise Manager Database Control 的数据库实例需要 1 GB
  - ASM 实例和 Oracle Restart 需要 1.5 GB
- 磁盘空间要求：
  - 3 GB 交换空间（基于 2 GB RAM）
  - 在 /tmp 目录中保留 1 GB 的磁盘空间
  - Oracle DB 软件需要 3.8 GB
  - Oracle Grid Infrastructure 软件需要 4.5 GB
  - 预配置数据库需要 1.7 GB（可选）
  - 快速恢复区需要 3.4 GB（可选）
- 操作系统（请参阅有关文档）



**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### Oracle Grid Infrastructure 和 Oracle DB 安装：系统要求

- 在 RAM 大于等于 1 GB、交换空间大于等于 1.5 GB 的计算机上，可以完成标准数据库安装。用于独立服务器的 Oracle Grid Infrastructure 标准安装还至少需要 1.5 GB RAM。
- 所需的交换空间大小取决于 RAM 大小（例如，对于 2 GB RAM，您需要 3 GB 交换空间）。有关详细信息，请参阅 OS 特定的安装指南。
- 标准安装可以在 20 分钟或更短时间内完成，具体取决于安装 Oracle DB 软件的计算机的活动程度。
- 有关安装的一些详细信息：
  - Oracle Database 11g 提供了两个种子数据库模板。
  - 将删除重复文件。
  - 大量其它产品和演示可以从附加的 CD 中安装。

本幻灯片中列出的硬件要求是所有平台的最低要求。您的具体安装可能还要满足其它要求（特别是磁盘空间）。

注：包括一个标准种子数据库的企业版安装类型称为“标准安装”。

## 准备操作系统

创建必需的操作系统用户和组:

- 组:
  - oinstall
  - dba
  - 可选组（如果要在多个用户间划分职责）:
    - oper
    - asmdba
    - asmoper
    - asmadmin
- 用户:
  - 软件所有者，通常为 oracle
  - 可以为多个产品安装创建多个用户



版权所有 © 2010, Oracle。保留所有权利。

### 准备操作系统

对于 Oracle 安装所用的硬件，系统管理员需要执行一些步骤。本课程不讨论所有操作系统的配置细节和命令。此外，每个操作系统对 Oracle 软件都有特定的要求。因此本课程将讨论一些概括性的要求，建议您参考特定操作系统的文档获取有关 Oracle 安装的信息。所需步骤之一是创建必要的操作系统组和用户。必需的两个操作系统组是: oinstall 和 dba。如果要在多个用户帐户间实施职责分离，则还应另外创建以下几个组: oper、asmdba、asmoper 和 asmadmin。至少需要有一个操作系统用户作为 Oracle 安装的所有者。大多情况下，配置 oracle 用户就是为了此目的。如果想实现真正的职责分离，可以为每个 Oracle 产品设置不同的所有者。

# 设置环境变量

## Oracle 环境变量：

- **ORACLE\_BASE**: Oracle 目录结构的基目录。建议在安装之前设置此项。
- **ORACLE\_HOME**: Oracle 产品运行环境。如果 ORACLE\_BASE 已设置，安装之前不一定要设置此项。
- **ORACLE\_SID**: 安装之前不一定要设置此项，但对简化以后与特定实例的交互很有用。
- **NLS\_LANG**: 可选环境变量，用于控制语言、地区和客户机字符集设置。



**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## 设置环境变量

在每个 Oracle 环境中都有很多 Oracle 环境变量。此处提到的只是对成功安装和使用 Oracle DB 很重要的那些变量。所有这些变量都不一定要进行设置，但是对其进行设置可避免将来出问题。

- **ORACLE\_BASE**: 按照 Oracle 技术支持部门所建议的最佳灵活体系结构 (OFA)，指定 Oracle 目录结构的基目录。此变量可选；但使用此变量可加快将来的安装和升级。它是一个目录路径，如下面的示例所示：  
`/u01/app/oracle`
- **ORACLE\_HOME**: Oracle 产品运行环境。如果 ORACLE\_BASE 已设置，安装之前不一定要设置此项。OUI 可在安装过程中根据 ORACLE\_BASE 的设置来确定并建议 ORACLE\_HOME 的设置。设置此环境变量，可以更轻松地维护和管理 Oracle 软件。它是一个目录路径，如下面的示例所示：  
`/u01/app/oracle/product/11.2.0/dbhome_1`
- **ORACLE\_SID**: Oracle 实例的系统标识符，例如 `orcl` 标识数据库，`+ASM` 标识 ASM 实例。安装之前不一定要设置此项，但对简化以后与特定实例的交互很有用。
- **NLS\_LANG**: 可选环境变量，用于控制语言、地区和客户机字符集设置，如下面的示例所示：  
`AMERICAN_DENMARK.WE8MSWIN1252`

## 设置环境变量（续）

有关有效的语言、地区、字符集和语言支持的详细信息，请参阅《Oracle Database Globalization Support Guide》。

## 检查系统要求

- 足够的临时空间
- 64 位与 32 位问题
- 正确的操作系统 (OS)
- OS 补丁程序级别
- 系统程序包
- 系统和内核参数
- X Server 权限
- 足够的交换空间
- ORACLE\_HOME 状态

```
[oracle@edrsrl2p1-+ASM Disk1]$ ./runInstaller
Starting Oracle Universal Installer...

Checking Temp space: must be greater than 80 MB.   Actual 15067 MB  Passed
Checking swap space: must be greater than 150 MB.  Actual 4000 MB  Passed
Checking monitor: must be configured to display at least 256 colors.  Actual 6
5536  Passed
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2009-05-15_12
:04-10AM. Please wait ...
```

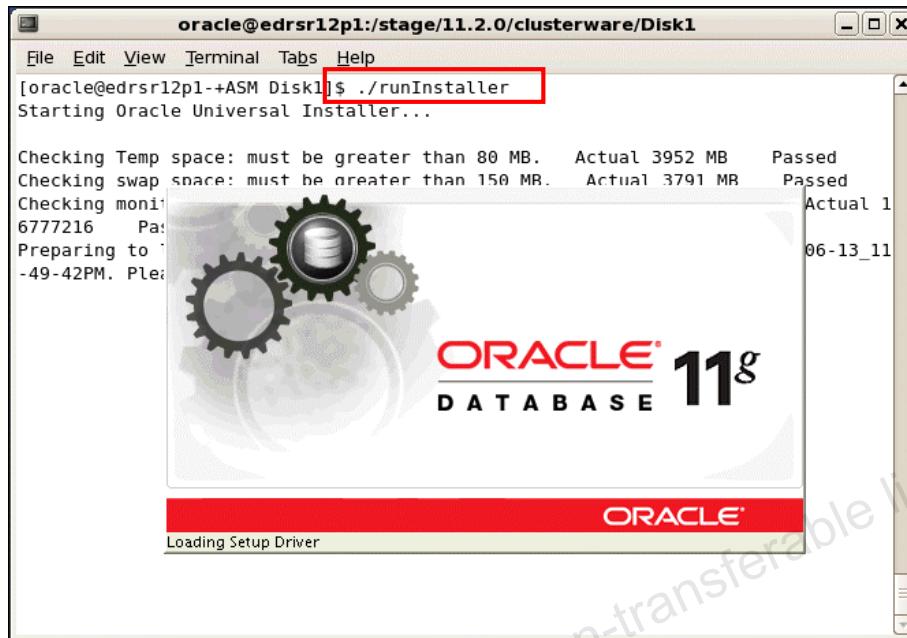
版权所有 © 2010, Oracle。保留所有权利。

### 检查系统要求

Oracle Universal Installer 自动运行大部分先决条件检查以验证下列项：

- 检查安装和配置所需的最小临时空间。在安装过程中会验证这些要求。
- 不允许将 64 位安装软件安装在安装了 32 位软件的 Oracle 主目录中（反之亦然）。
- 确定 Oracle Grid Infrastructure 11g 和 Oracle Database 11g 的安装平台已经过认证，目前有几个 Linux 平台版本及其它平台已经过认证。
- 已安装所有必需的 OS 补丁程序。
- 正确设置了所有必需的系统和内核参数。
- 设置了 DISPLAY 环境变量，并且用户有足够的权限在指定的 DISPLAY 下显示信息。
- 系统具有充足的交换空间。
- Oracle 主目录要么是空的（新安装），要么是可以安装 Oracle Database 11g 的几个支持版本之一。安装过程中还会验证这些版本是否在 Oracle 产品清单中进行了注册。

## Oracle Universal Installer (OUI)



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle Universal Installer (OUI)

Oracle Universal Installer (OUI) 是一个 Java 应用程序，它可以基于组件执行安装，支持不同级别的基于集成绑定、套件和 Web 的安装，并且支持在单个程序包中包含复杂逻辑。安装引擎可以方便地在所有支持 Java 的平台上移植，可以封装整个安装过程中特定于平台的问题。

OUI 为进行软件管理和分配提供以下功能：

- 自动的依赖性解析和复杂逻辑处理
- 从 Web 安装
- 组件和套件安装
- 隐式卸载
- 支持多个 Oracle 主目录
- NLS 或全球化支持
- 支持分布式安装
- 使用响应文件的无人值守“无提示”安装

## 示例：安装场景

本课中演示的安装场景分为两部分：

- 第一部分：安装用于独立服务器的 Oracle Grid Infrastructure
- 第二部分：安装 Oracle DB 软件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 示例：安装场景

本课中演示的安装场景分为两部分：

- 第一部分：安装用于独立服务器的 Oracle Grid Infrastructure
- 第二部分：安装 Oracle DB 软件

在 Oracle Grid Infrastructure 安装中，将演示配置 ASM 磁盘组的步骤，并且将配置 Oracle Restart。首先安装 Oracle Grid Infrastructure，这样安装 Oracle DB 软件后创建的数据库将可以使用 ASM 磁盘组，并且可以在 Oracle Restart 中自动注册。

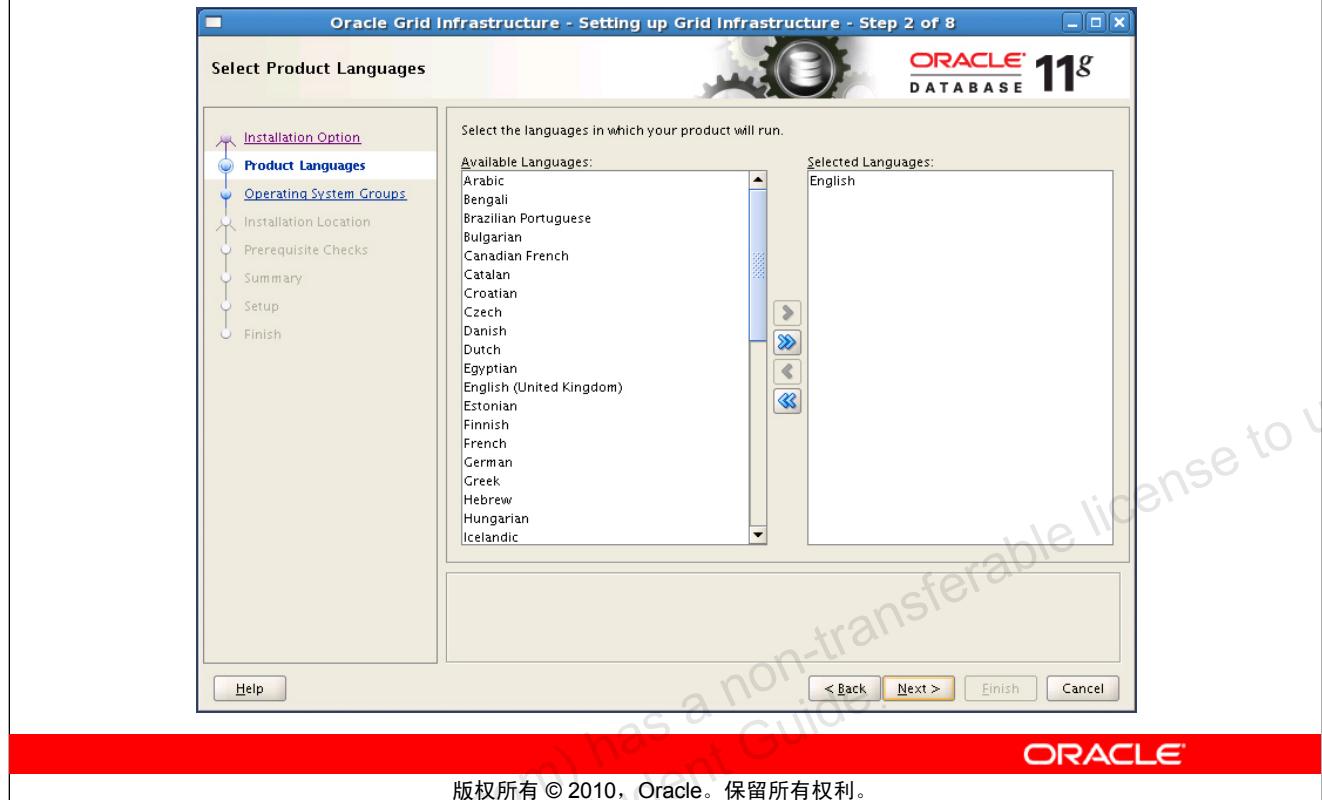
# 第一部分：安装用于独立服务器的 Oracle Grid Infrastructure



## 第一部分：安装用于独立服务器的 Oracle Grid Infrastructure

要使用 Oracle Universal Installer (OUI) 安装 Oracle Grid Infrastructure 软件，请以管理组成员身份登录计算机，且该管理组已被授权允许安装 Oracle 软件以及创建和管理数据库。将集群件的发行 CD 插入到 CD 驱动器，或者导航至 Oracle 集群件的存放位置。在该位置输入 `./runInstaller` 启动 OUI。此时会出现 OUI 的“Installation Option（安装选项）”页。选择“Install and Configure Grid Infrastructure for a Standalone Server（安装并配置用于独立服务器的 Grid Infrastructure）”选项，然后单击“Next（下一步）”。

## 选择产品语言



### 选择产品语言

此时将显示“Select Product Languages（选择产品语言）”页。要将一种语言添加到安装中，单击该语言将其突出显示，然后使用向右箭头按钮将其移至“Selected Languages（所选语言）”列表。按住 Ctrl 键的同时，单击鼠标可以选择多种语言。单击“Next（下一步）”按钮继续安装。

当前的语言列表包括：Arabic（阿拉伯语）、Bengali（孟加拉语）、Brazilian Portuguese（巴西葡萄牙语）、Bulgarian（保加利亚语）、Canadian French（加拿大法语）、Catalan（加泰罗尼亚语）、Croatian（克罗地亚语）、Czech（捷克语）、Danish（丹麦语）、Dutch（荷兰语）、Egyptian（埃及语）、English (United Kingdom)（英国英语）、Estonian（爱沙尼亚语）、Finnish（芬兰语）、French（法语）、German（德语）、Greek（希腊语）、Hebrew（希伯来语）、Hungarian（匈牙利语）、Icelandic（冰岛语）、Indonesian（印度尼西亚语）、Italian（意大利语）、Japanese（日语）、Korean（朝鲜语）、Latin American Spanish（拉丁美洲西班牙语）、Latvian（拉脱维亚语）、Lithuanian（立陶宛语）、Malay（马来语）、Mexican Spanish（墨西哥西班牙语）、Norwegian（挪威语）、Polish（波兰语）、Portuguese（葡萄牙语）、Romanian（罗马尼亚语）、Russian（俄语）、Simplified Chinese（简体中文）、Slovak（斯洛伐克语）、Slovenian（斯洛文尼亚语）、Spanish（西班牙语）、Swedish（瑞典语）、Thai（泰语）、Traditional Chinese（繁体中文）、Turkish（土耳其语）、Ukrainian（乌克兰语）和 Vietnamese（越南语）。

**注：**此列表会随着更新而发生变化。



## 创建 ASM 磁盘组

此时将显示“Create ASM Disk Group（创建 ASM 磁盘组）”页。Oracle Grid Infrastructure 支持 ASM 和 Oracle Restart。只有创建了 ASM 磁盘组后，OUI 才会继续运行。在“Disk Group Name（磁盘组名）”字段中输入要创建的第一个 ASM 磁盘组的名称。如果“Add Disks（添加磁盘）”部分未填入候选磁盘，请单击“Change Discovery Path（更改搜索路径）”按钮，然后输入 ASM 磁盘的磁盘搜索路径。

对于大多数环境，Oracle 建议创建两个磁盘组。但是，此时 OUI 只能创建一个 ASM 磁盘组。您可以在安装结束后使用 ASM Configuration Assistant (asmca) 实用程序或 SQL\*Plus 创建推荐的第二个 ASM 磁盘组。单击“Next（下一步）”按钮继续安装。

**注：**示例中的列表之所以显示这些设备，是因为配置了 ASMLib。ASMLib 只适用于 Linux 平台。在其它平台上，可能需要单击“Change Discovery Path（更改搜索路径）”按钮。



## 定义 ASM 口令

此时将显示“Specify ASM Password（指定 ASM 口令）”页。必须提供两个帐户的口令：SYS 帐户和 ASMSNMP 帐户。您可以选择对这些帐户使用不同的口令，也可以选择使用同一个口令。与 Oracle DB 不同，ASM 实例不含数据字典，因此只有操作系统验证和口令文件验证这两种验证方法。SYS 帐户会添加到口令文件中（在 Linux 上为 orapw+ASM），并授予 SYSDBA、SYSOPER 和 SYSASM 权限。ASMSNMP 帐户会添加到口令文件中，但只授予 SYSDBA 权限。输入相应的口令后，单击“Next（下一步）”按钮继续安装。



## 定义已授权的操作系统组

此时将显示“Privileged Operating System Groups（已授权的操作系统组）”页。如果当前用户是以下组的操作系统成员，OUI 实用程序将为这些组提供建议的默认值：

- ASM 数据库管理员 (OSDBA) 组 – asmdba
- ASM 实例操作员 (OSOPER) 组 – asmoper
- ASM 实例管理员 (OSASM) 组 – asmadmin

因为此安装针对的是独立服务器，因此通常的做法是三个组都使用同一个操作系统组，如 dba（如幻灯片中所示）。单击“Next（下一步）”按钮继续安装。由于 OSDBA、OSOPER 和 OSASM 使用同一个操作系统组，所以会显示警告，对此请单击“Yes（是）”。



## 指定安装位置

此时将显示“Specify Installation Location（指定安装位置）”页。对于“Oracle Base（Oracle 基目录）”字段，输入软件所有者的 ORACLE\_BASE 的值。默认值为 /u01/app/oracle。对于“Software Location（软件位置）”字段，输入 Grid Infrastructure 软件的 ORACLE\_HOME 的值。默认值为 /u01/app/oracle/product/11.2.0/grid。单击“Next（下一步）”按钮继续安装。



### 创建清单

如果主机上还没有 Oracle 主产品清单目录，则将显示“Create Inventory（创建清单）”页。对于“Inventory Directory（清单目录）”字段，为 oraInventory 目录输入所需位置。推荐的目录为 /u01/app/oraInventory。oraInventory 目录应该比 ORACLE\_BASE 目录高一级。从选择列表中选择 oraInventory 组名称。推荐的名称为 oinstall。单击“Next（下一步）”按钮继续安装。



## 执行先决条件检查

此时将显示“Perform Prerequisite Checks（执行先决条件检查）”页。除非由于出错而未通过某些检查，否则无需在此页上执行任何操作。如果确实出错，需要在继续安装之前将其纠正。在幻灯片示例中，有一些内核参数没有通过先决条件检查。系统指示您运行一个脚本来纠正这些内核参数，然后重新运行检查。如果是无法修复的错误，您必须手动修复该问题或者选择“Ignore All（全部忽略）”，然后继续安装（如果未通过的那项检查不是关键检查）。

**注：**并非所有故障都能使用 OUI 生成的 `runfixup.sh` 脚本来修复。在这种情况下，需要手动干预来纠正问题，然后才能重新尝试安装。



## 验证安装概要数据

此时将显示“Summary（概要）”页。复查呈现的信息，这些信息与在之前页面中提供的安装答案相关。将会出现将交互安装问题及答案保存至响应文件的选项。响应文件可用于在将来的安装中执行无提示安装，其安装方式与正在执行的此安装相同。响应文件是一个文本文件，可以使用文本编辑工具进行编辑，以便针对不同的安装进行修改。单击“Finish（完成）”按钮继续安装。



## 监视安装进度

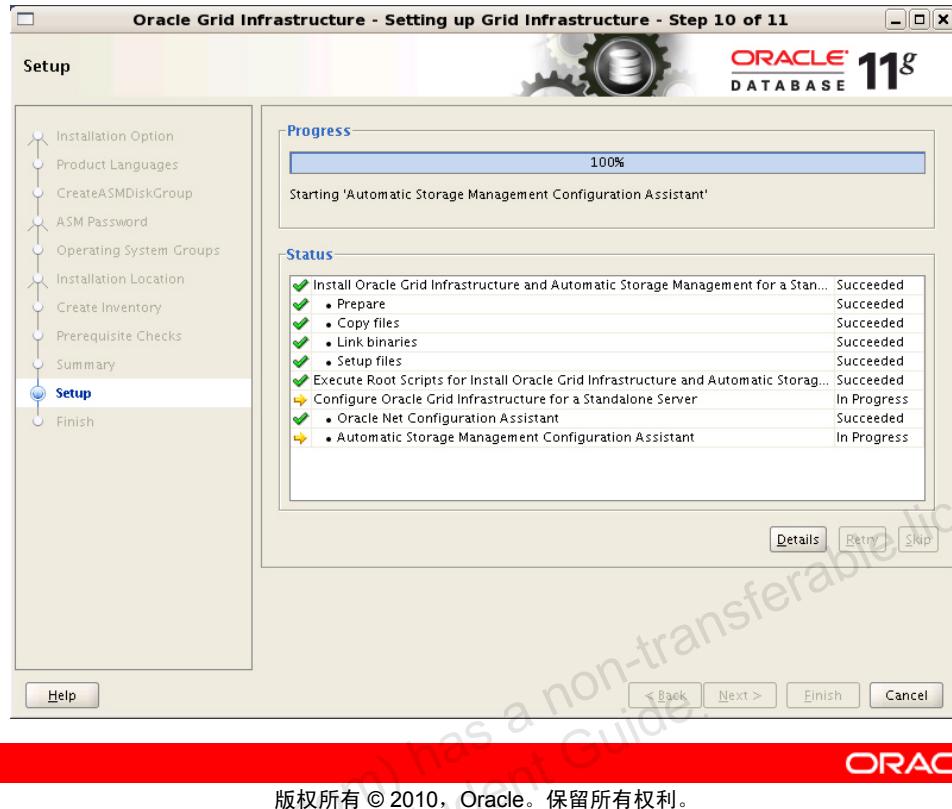
此时将出现“Setup（安装）”页，显示安装的进度。进度包括准备安装、复制文件、链接库和创建安装文件。除非由于出错而安装失败，否则无需在此页上执行任何操作。如果确实出错，需要在继续安装之前将其纠正。



## 执行 root 配置脚本

接下来将显示“Execute Configuration Scripts（执行配置脚本）”对话框页，显示需要以 root 用户身份执行的配置脚本。`orainstRoot.sh` 脚本负责更改 Oracle 主产品清单目录的权限，即添加组的读写权限，同时删除 world 用户的读写和执行权限。`root.sh` 脚本将文件复制到 `/usr/local/bin` 目录中，创建 `/etc/oratab` 文件，创建 grid 用户的 OCR 注册表项，启动 `ohasd` 守护程序，并修改 `/etc/inittab` 以便在计算机启动时自动启动 `ohasd` 守护程序。执行 root 脚本后，单击“OK（确定）”按钮返回到“Setup（安装）”进度页继续安装。

# 执行 Configuration Assistant



## 执行 Configuration Assistant

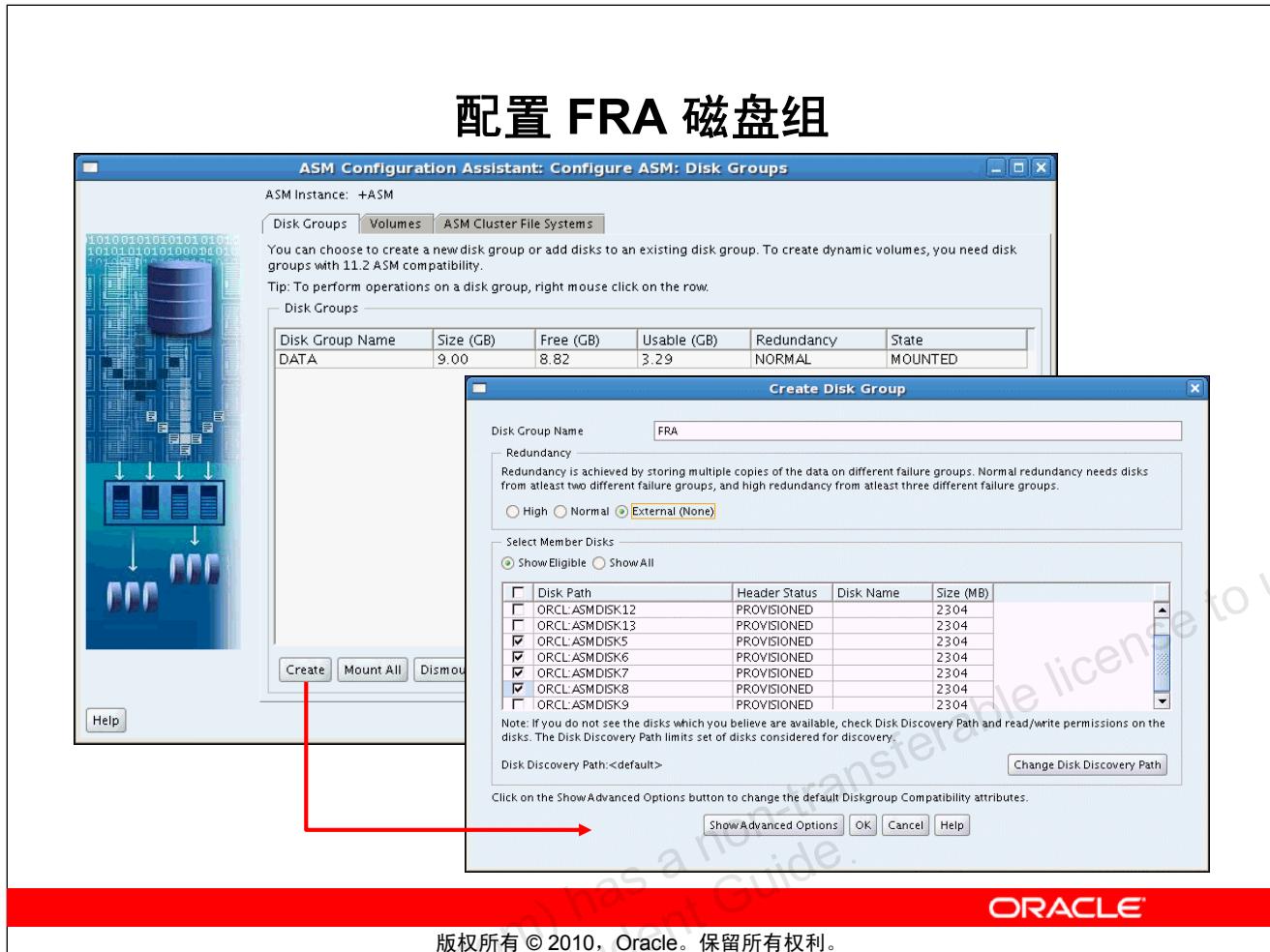
采用无提示安装方法时，将调用 Oracle Net Configuration Assistant (netca) 来创建网络配置文件，随后 Automatic Storage Management Configuration Assistant (asmca) 将创建 ASM 磁盘组并在 Oracle Restart 中注册 ASM 组件。

# 完成安装



## 完成安装

安装结束以后，将显示“Finish（完成）”页。单击“Close（关闭）”按钮退出 OUI 实用程序。



## 配置 FRA 磁盘组

由于在 Oracle Grid Infrastructure 安装过程中只能配置一个磁盘组，因此必须手动创建其它磁盘组。在此场景中，我们需要一个可以用作数据库快速恢复区的 FRA 磁盘组。ASM Configuration Assistant (asmca) 实用程序提供了一个直观的 GUI 界面，通过该界面可以轻松地创建新的 ASM 磁盘组或删除现有的 ASM 磁盘组。

## 小测验

Universal Installer 执行安装 Oracle 软件所必需的所有配置工作。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

**答案: 2**

**小测验注释**

安装之前必须在操作系统（例如 UNIX 和 Linux）上创建操作系统用户和组。

## 小测验

在 Grid Infrastructure 安装过程中，可以执行的操作是：

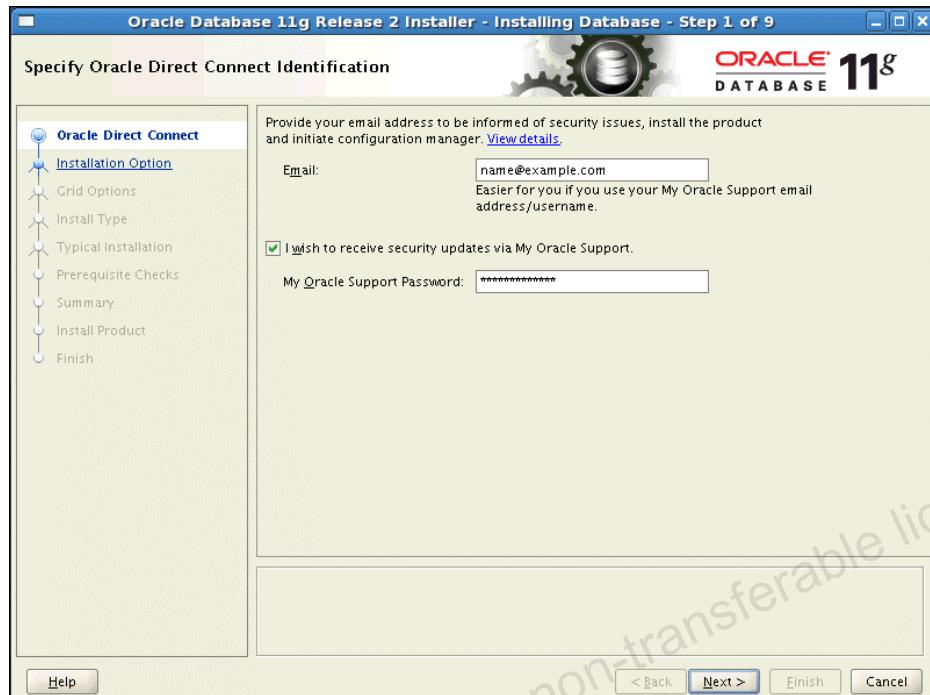
1. 指定 ASM 实例数据文件的确切位置
2. 只能创建一个磁盘组
3. 指定 ASM 实例的 SGA 大小
4. 创建多个磁盘组

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2

## 第二部分：安装 Oracle DB 软件



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 第二部分：安装 Oracle DB 软件

现在，我们来看一下 Oracle DB 软件的安装步骤。和之前一样，请以管理组成员身份登录计算机，且该管理组已被授权允许安装 Oracle 软件以及创建和管理数据库。将数据库发行 CD 插入 CD 驱动器中，或者导航至 Oracle DB 存放位置。在该位置输入 `./runInstaller` 来启动 Oracle Universal Installer (OUI)。如果需要，输入您希望用于接收安全问题通知的电子邮件地址。如果希望通过 My Oracle Support 接收安全更新，请同时输入您的 My Oracle Support 口令。如果不输入电子邮件地址，则会显示一条警告消息，询问您是否确实不想接收关于配置的重要问题的通知。对此警告消息，单击“Yes (是)”，继续安装。



## 选择安装类型

此时将显示“Select Installation Option（选择安装选项）”页。选择要执行的安装类型，然后单击“Next（下一步）”：

- **Create and Configure a Database**（创建并配置数据库）：此选项将在安装产品之后创建一个数据库。
- **Install Database Software Only**（仅安装数据库软件）：此选项将仅安装 Oracle DB 二进制文件。
- **Upgrade an Existing Database**（升级现有数据库）：此选项用于升级较早版本的数据库。

## 选择网格安装选项



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 选择网格安装选项

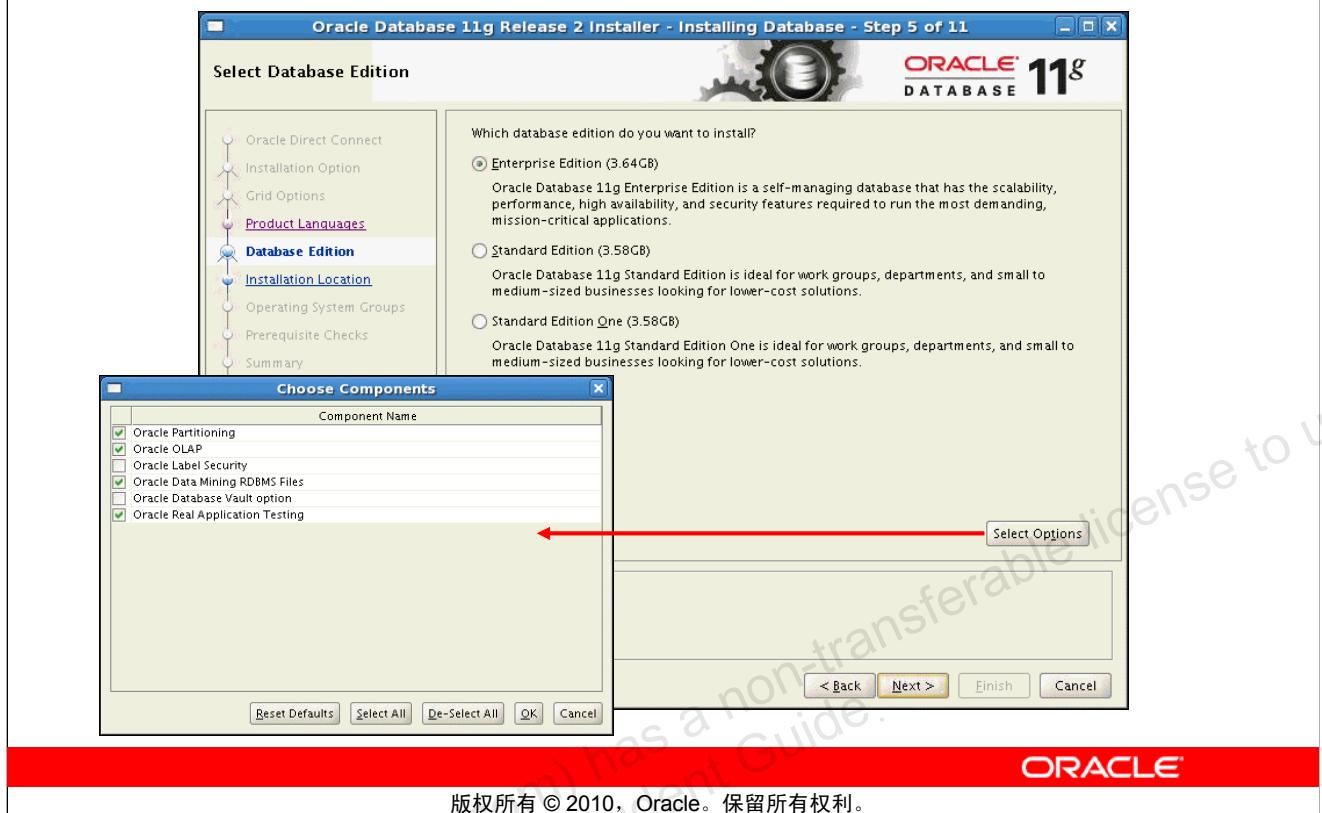
此时将显示“Grid Installation Option（网格安装选项）”页。选择是进行单实例数据库安装，还是在集群上进行 Oracle RAC 数据库安装。单击“Next（下一步）”继续。



## 选择语言设置

接下来将显示“Select Product Languages（选择产品语言）”页。在此页可以选择产品运行时使用的全部语言。英语是默认语言，并且不能从“Selected Languages（所选语言）”列表中删除。单击“Next（下一步）”继续。

## 选择数据库版本



### 选择数据库版本

此时将显示“Select Database Edition（选择数据库版本）”页。可以选择以下三个版本之一：

- **Enterprise Edition（企业版）**：此版本提供一个自助管理数据库，该数据库根据选择的是集群服务器还是单一服务器来提供性能、可伸缩性、安全性和可靠性。其提供了全面的功能，可以轻松管理要求极苛刻的事务处理、业务智能和内容管理应用程序。
- **Standard Edition（标准版）**：此版本提供一个全功能数据库，支持最多带四个插槽的服务器。该版本包括具有更高可用性的 Oracle Real Application Clusters，并提供企业级的性能和安全性，易于管理，并且可以随需求的增长轻松地扩展。并且可以与企业版向上兼容。
- **Standard Edition One（标准版一）**：此版本提供一个全功能数据库，支持最多带两个插槽的服务器。它提供企业级的性能和安全性，易于管理，并且可随需求的增长轻松地扩展。并且可以与其它数据库版本向上兼容。

单击“Select Options（选择选项）”按钮进一步定制要安装的组件。完成“Select Database Edition（选择数据库版本）”页上的所有选项后，单击“Next（下一步）”。



## 指定安装位置

接下来将显示“Specify Installation Location（指定安装位置）”页。默认显示推荐的 Oracle 基目录路径。可以根据需要更改该路径。在“Software Location（软件位置）”部分，可以接受默认值，也可以输入您希望安装 Oracle 组件的 Oracle 主目录名称和目录路径。目录路径不能包含空格。单击“Next（下一步）”继续完成安装过程。



## 选择操作系统组

此时将显示“Privileged Operating System Groups（已授权的操作系统组）”页。为 OSDBA 和 OSOPER 权限选择合适的操作系统组。默认情况下，OSDBA 指定为 dba，OSOPER 指定为 oper。在本例中，两个组都使用 dba，这是因为我们进行的是不实施职责分离的独立安装。单击“Next（下一步）”继续。

## 执行先决条件检查



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 执行先决条件检查

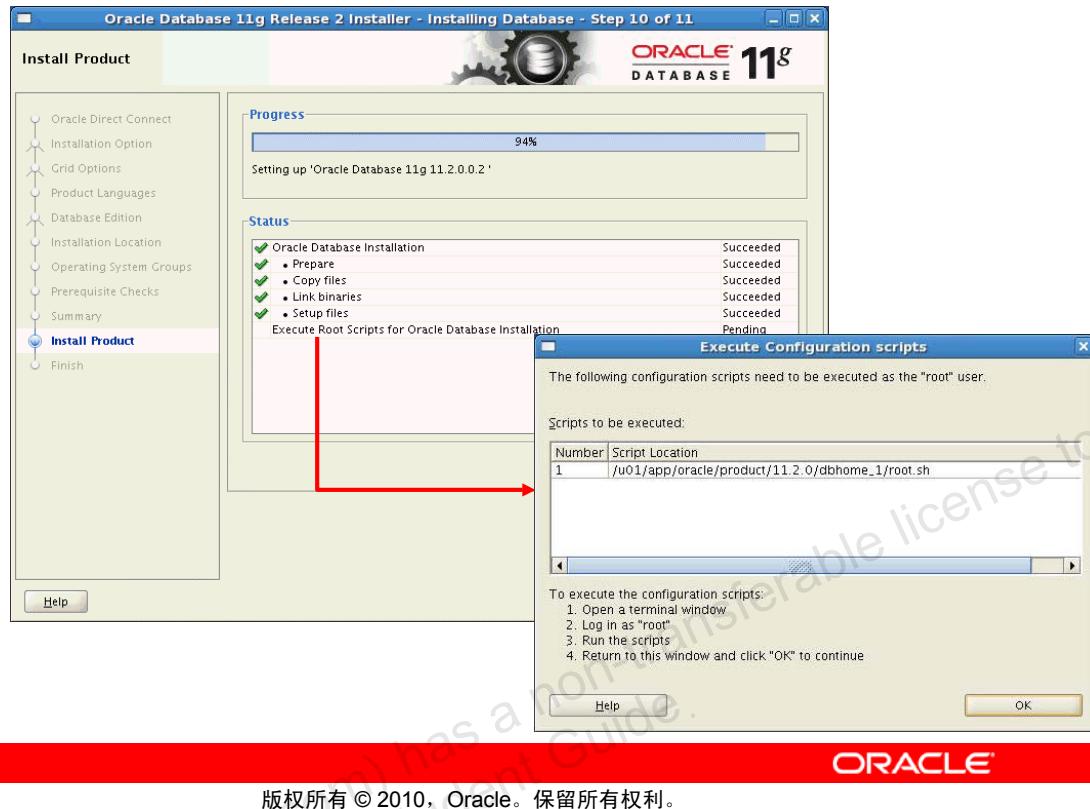
此时将显示“Perform Prerequisite Checks（执行先决条件检查）”页。OUI 逐一执行多项先决条件检查。执行了全部检查以后，如果有任何测试失败，OUI 将返回有关信息。除非由于出错而未通过某些检查，否则无需在此页上执行任何操作。如果有任何先决条件检查失败，则会显示一个页面。您可以在该页面中单击“Fix & Check Again（修复并重新检查）”，这时 OUI 将生成一个脚本来修复其能够修复的所有问题。如果是 OUI 无法修复的错误，您必须手动修复该问题，或者选择“Ignore All（全部忽略）”，然后继续安装（如果未通过的那项检查不是关键检查）。



## 安装概要页

此时将显示“Summary（概要）”页。复查呈现的信息，这些信息与在之前页面中提供的安装答案相关。将会出现将交互安装问题及答案保存至响应文件的选项。响应文件可用于在将来的安装中执行无提示安装，其安装方式与正在执行的此安装相同。响应文件是一个文本文件，可以使用文本编辑工具进行编辑，以便针对不同的安装进行修改。单击“Finish（完成）”按钮继续安装。

## 安装产品页



### 安装产品页

安装过程在此处暂停，要求您以 root 用户身份执行一个附加的配置脚本。在另一个终端窗口中输入：

```
$ su
# password: oracle <root 口令，不会在窗口中显示>
# /u01/app/oracle/product/11.2.0/dbhome_1/root.sh
```

如果是在 Linux 或 UNIX 上进行安装，请接受默认的本地 bin 目录。脚本完成后，从 root 帐户退出并关闭窗口，然后在“Execute Configuration scripts（执行配置脚本）”对话框中单击“OK（确定）”以允许安装完成。

# 安装完成



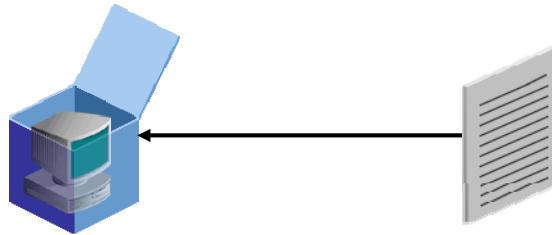
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 安装完成

完成所有的安装步骤后，将显示“Finish（完成）”页。单击“Close（关闭）”按钮退出OUI实用程序。

## 安装选项：无提示模式



要在无提示模式下使用 OUI 安装并配置 Oracle 产品，请执行以下步骤：

1. 准备响应文件：

- 使用随 Oracle 软件提供的文件模板之一
- 在使用 OUI 进行安装过程中记录响应文件，方法是在概要页上单击“Save Response File（保存响应文件）”

2. 在无提示模式或隐藏模式下运行 OUI。

```
./runInstaller -silent -responsefile <filename>
```

如果需要，在无提示模式下运行 NetCA 和 DBCA。

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## 安装选项：无提示模式

要在无提示或隐藏模式下使用 OUI 安装并配置 Oracle 产品，请执行以下操作：

1. 准备响应文件。可以使用为每个产品和安装类型提供的文件模板之一（例如 enterprise.rsp、standard.rsp 和 netca.rsp）来准备此文件。如果这些模板不适合您的方案，也可以在交互模式下使用 OUI 记录响应文件。您可以编辑该文件，然后再使用它完成无提示模式或隐藏模式安装。在使用 OUI 进行交互式安装的过程中，单击概要页上的“Save Response File（保存响应文件）”可完成响应文件的记录。
2. 在无提示模式或隐藏模式下运行 OUI。在安装结束时运行 \$ORACLE\_BASE/oraInventory/orainstRoot.sh 和 \$ORACLE\_HOME/root.sh。如果完成了仅限软件的安装，则可根据需要在无提示模式或非交互模式下运行 Oracle Net Configuration Assistant (NetCA) 和 Database Configuration Assistant (DBCA)。

有关详细信息，请参阅特定于操作系统的《Oracle 数据库安装指南》。

## 小测验

响应文件是：

1. 可使用二进制编辑程序编辑的二进制文件
2. 可由安装程序创建的二进制文件
3. 不能编辑但可由安装程序创建的文本文件
4. 可使用文本编辑器编辑的文本文件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：4

## 小测验

在安装数据库软件过程中，可以为其指定组的是：

1. osoper 组
2. osasm 组
3. osdba 组
4. osadmin 组

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：1、3

## 小结

在本课中，您应该已经学会：

- 描述数据库管理员 (DBA) 的角色，说明典型的任务和工具
- 计划 Oracle 软件安装
- 安装用于独立服务器的 Oracle Grid Infrastructure
- 安装 Oracle DB 软件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 2 概览：准备数据库环境

本练习包括使用 Oracle Universal Installer 安装 Oracle 软件。

**注：完成本练习对后面的所有练习来说至关重要。**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



## 使用 DBCA 创建 Oracle DB

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成下列工作：

- 使用 Database Configuration Assistant (DBCA) 创建数据库
- 使用 DBCA 生成数据库创建脚本
- 使用 DBCA 管理数据库设计模板
- 使用 DBCA 执行其它任务

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

# 计划数据库

作为 DBA，必须计划：

- 数据库的逻辑存储结构及其物理实施：
  - 您有多少磁盘驱动器？使用何种类型的存储？
  - 需要多少个数据文件？（计划要考虑未来增长。）
  - 要使用多少表空间？
  - 要存储哪些类型的信息？
  - 是否存在由类型或大小产生的任何特殊存储要求？
- 整体数据库设计
- 数据库备份策略



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 计划数据库

计划数据库逻辑存储结构时，需要考虑其会怎样影响系统性能和各种数据库管理操作，这是一项非常重要的工作。例如，在创建数据库的任何表空间之前，应该知道由多少个数据文件构成表空间，在每一个表空间中存储何种类型的信息，以及数据文件实际会存储在哪些磁盘驱动器上。网络连接存储 (NAS) 的可用性以及专用存储网络的带宽等信息十分重要。如果要使用存储区域网络 (SAN)，则知道逻辑卷的配置方式以及条带大小是很有帮助的。

计划数据库的整体逻辑存储结构时，应考虑此结构在实际创建和运行数据库时会产生影响。某些数据库对象可能由于类型或大小而有特殊的存储要求。

在分布式数据库环境中，此计划阶段是一个极其重要的阶段。存放频繁访问的数据的物理位置对应用程序性能有很大的影响。

在计划阶段，可以制定数据库的备份策略。您可以变更数据库的逻辑存储结构或设计来提高备份效率。备份策略将在后面的课中进行介绍。

## 数据库：示例

- 通用数据库和事务处理数据库：
  - 联机事务处理 (OLTP) 系统，例如软件公司或苗圃的零售记帐系统
- 定制：
  - 多用途数据库（可能结合了 OLTP 和数据仓库功能）
- 数据仓库：
  - 研究和市场营销数据
  - 省/州税或联邦纳税数据
  - 专业人员认证（医生、护士等）



版权所有 © 2010, Oracle。保留所有权利。

### 数据库：示例

不同类型的数据库具有各自特定的实例和存储要求。Oracle DB 软件包含一些用于创建不同类型数据库的模板。这些典型示例包括：

- **通用**：满足一般或事务处理需要，例如进行事务处理并将其存放长度适中的一段时间
- **定制**：不宜使用通用或数据仓库模板的定制数据库
- **数据仓库**：需要长期存储数据并通过读操作检索数据的数据库

本页和上一页中的信息提供了 DBA 需要考虑的注意事项。本课程的整个内容旨在帮助您解决这些问题。

## 选择适当的字符集

- Oracle DB 支持多种字符编码方案：
  - 单字节字符集
    - 7 位
    - 8 位
  - 多字节字符集，包括 Unicode
- 字符集是在创建数据库时选择的。请选择最能满足您现在和将来业务需要的字符集，因为选定字符集后很难再更改。
- 一般情况下，建议选择 Unicode，因为它是最灵活的字符集。



版权所有 © 2010, Oracle。保留所有权利。

### 选择适当的字符集

处理字符时，计算机系统使用的是数字代码，而不是以图形方式表示的字符。“编码字符集”将数字代码映射到计算机或终端可以显示和接收的字符。不同的字符集支持不同的字符指令表。由于字符集通常基于特定的编写脚本，因此，它们可以支持多种语言。但是，基于脚本的字符集只支持基于类似脚本的几组语言，从这个意义上而言，这类字符集是有限的。通用字符集囊括了现代绝大多数主要脚本，为多语言支持提供了更为有用的解决方案。有关 Unicode 标准的信息，请参阅相关网站 <http://www.unicode.org>。

Oracle DB 支持三种编码方案：单字节方案、宽度可变的多字节方案以及通用方案。请选择最能满足您现在和将来业务需要的正确字符集，因为选定字符集后很难再更改。为了获得最佳性能，选择的字符集不但要避免字符集的转换，而且要使用对所需的语言而言最高效的编码。与多字节字符集相比，单字节字符集的性能更佳。此外，单字节字符集占用的空间最少。但是，单字符字符集所能支持的语言数量有限。为了选择正确的数据库字符集，请评估当前和未来的业务需要以及技术要求（例如，XML 和 Java 标准要求使用 Unicode）。一般来说，Oracle 建议您对所有新数据库使用 Unicode，因为 Unicode 是最灵活的字符集，可以避免以后转换。

## 选择适当的字符集（续）

### 单字节字符集

在单字节字符集中，每个字符占用一个字节。单字节 7 位编码方案最多可以定义 128 ( $2^7$ ) 个字符；单字节 8 位编码方案最多可以定义 256 ( $2^8$ ) 个字符。

### 单字节方案示例

7 位字符集：

- 7 位美国信息交换标准代码 (ASCII) (US7ASCII)

8 位字符集：

- 西欧国际标准组织 (ISO) 8859-1 (WE8ISO8859P1)
- 8 位西欧 DEC (WE8DEC)
- 8 位意大利扩展的二进制编码十进制交换码 (EBCDIC) 码页 1144 (I8EBCDIC1144)

### 多字节字符集

宽度可变的多字节字符集，其中每个字符用一个或多个字节表示。多字节字符集通常用来支持亚洲语言。某些多字节编码方案使用最高有效位值来指示：一个字节是表示单个字符，还是属于表示一个字符的一系列字节的一部分。但是，其它字符编码方案区分单字节字符与多字节字符。由设备发送的移出控制码指示，遇到移入代码之前，任何连续的字节都是双字节字符。涉及移入移出的编码方案主要用在 IBM 平台上。

Unicode 是一种通用编码字符集，支持使用单个字符集存储任何一种语言的信息。无论使用何种平台、程序或语言，Unicode 都能为每个字符提供唯一的代码值。

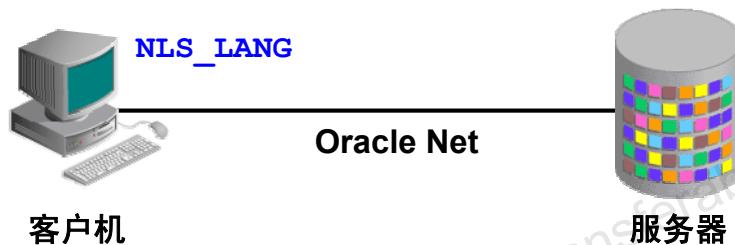
Unicode 标准已为许多软件和硬件供应商采用。现在，很多操作系统和浏览器都支持 Unicode。Unicode 是 XML、Java、JavaScript、LDAP 和 WML 等标准要求使用的字符集。它还与 ISO/IEC 10646 标准保持同步。

### 宽度可变的多字节方案示例

- 16 位日文 Shift-JIS (JA16SJIS)
- 使用香港增补字符集 HKSCS-2001 的 MS Windows 码页 950 (ZHT16HKSCS)
- Unicode 4.0 UTF-8 通用字符集 (AL32UTF8) — 一种宽度可变的编码类型，同时也是 ASCII 的严格超集
- Unicode (AL16UTF16) — Microsoft Windows 2000 和 Windows XP 使用的 16 位 Unicode 编码

## 如何使用字符集

- Oracle Net 将客户机上的 NLS\_LANG 设置与服务器上的字符集进行比较。
- 如果需要，会自动进行透明转换。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 如何使用字符集

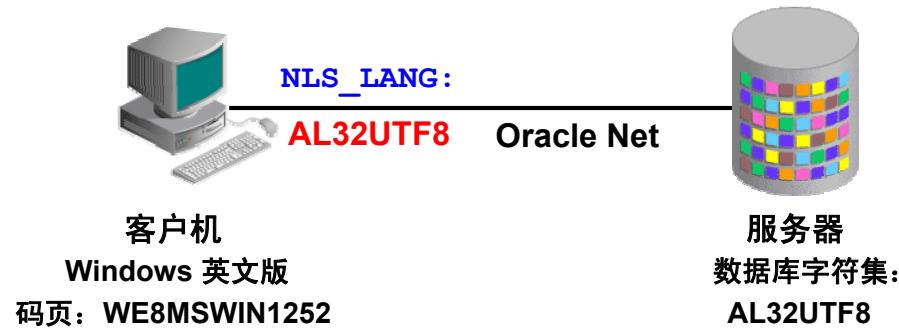
NLS\_LANG 参数定义了客户机终端的字符编码方案。不同的客户机可以使用不同的编码方案。客户机与服务器之间传递的数据可以在两种编码方案之间自动转换。数据库的编码方案应该是所有客户机编码方案的超集，或者与所有客户机编码方案等同。对于客户机应用程序，转换是透明的。

数据库字符集与客户机字符集相同时，数据库会假定正在发送或接收的数据属于同一字符集，因此不会执行验证或转换。

如果客户机应用程序所在的平台与服务器的不同，且该平台使用不同的字符编码方案，则在客户机/服务器环境中可能需要转换字符集。对于在客户机与服务器之间传递的字符数据，必须在两种编码方案之间进行转换。通过 Oracle Net，可以自动进行透明的字符转换。

## 要避免的问题

示例：



不发生转换，因为好像没有必要。

问题：在数据库中输入了无效数据。

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 要避免的问题

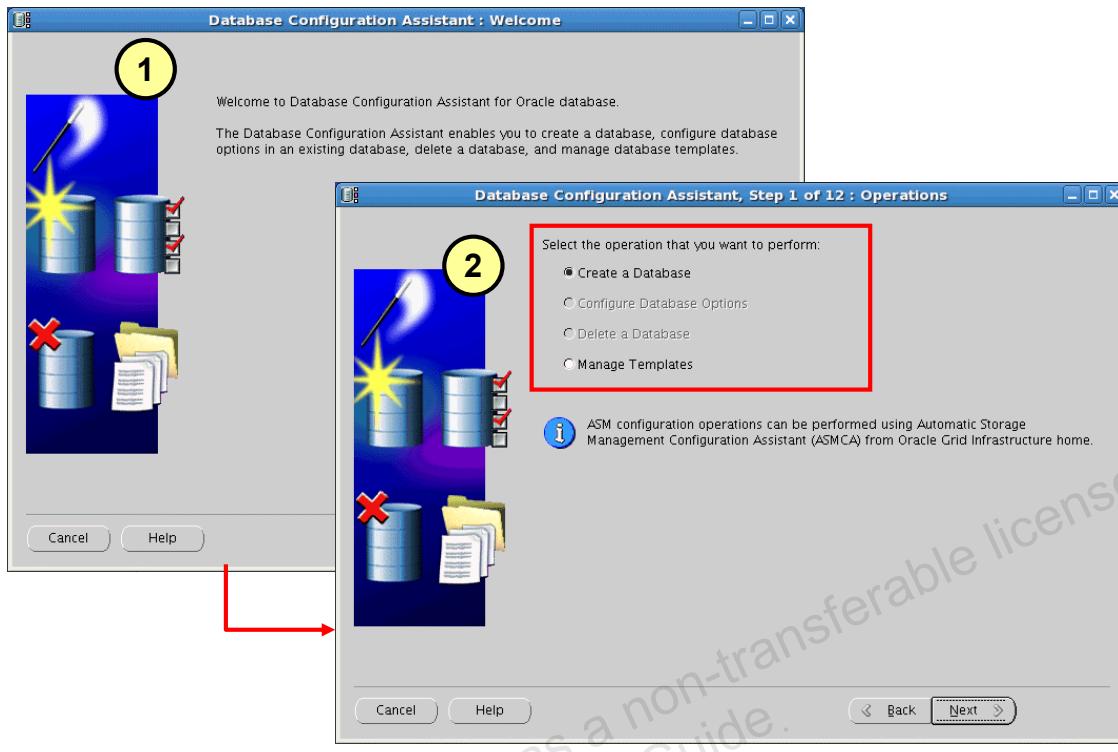
如果客户机上的 NLS\_LANG 参数未正确设置，则通常会在数据库中输入无效数据。

NLS\_LANG 值应该反映传入数据的编码。

- NLS\_LANG 参数的设置正确时，数据库可以自动转换从客户机操作系统传入的数据。
- NLS\_LANG 参数的设置错误时，将不会正确转换输入数据库的数据。

例如，假定数据库字符集是 AL32UTF8，客户机为英文版的 Windows 操作系统（码页为 WE8MSWIN1252），客户机上的 NLS\_LANG 设置为 AL32UTF8。输入数据库的数据使用 WE8MSWIN1252 编码，但没有转换为 AL32UTF8 数据，因为客户机上的 NLS\_LANG 设置与数据库字符集相符。所以，Oracle DB 认为无需进行转换，从而将无效的数据输入到数据库中。

## Database Configuration Assistant (DBCA)



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Database Configuration Assistant (DBCA)

Database Configuration Assistant (DBCA) 提供了多种选项来协助您执行各种操作。您可以使用 DBCA 创建或删除数据库，或者更改数据库的配置。还可以使用一系列预定义模板创建数据库，或者使用现有数据库作为样本来创建新数据库或模板。

DBCA 提供了多种选项，供您根据自己的需求创建数据库。DBCA 还提供了一系列页面，您可以在其中输入配置信息。在大多数页面中，DBCA 会提供默认设置。如果适用，您可以接受这些设置。使用 DBCA 创建数据库的步骤如下：

1. 以授权允许安装 Oracle 软件的 OS DBA 组成员身份登录计算机。如果需要，可以设置环境变量并输入 dbca 来调用 DBCA。此时将显示欢迎主页，您可以单击“Next (下一步)”继续。
2. 选择要执行的操作。在本例下，选择“Create a Database (创建数据库)”并单击“Next (下一步)”开始该操作。



## 使用 DBCA 创建数据库

3. **Database Templates (数据库模板)**：选择创建数据库时使用的数据库模板类型。数据库模板有三类，分别是“General Purpose or Transaction Processing（通用或事务处理）”、“Custom Database（定制数据库）”和“Data Warehouse（数据仓库）”。这些模板复制了预配置的数据库，其中包括数据文件。这些文件包括控制文件、重做日志文件以及所包括的各种表空间的数据文件。单击“Show Details（显示详细资料）”查看每类数据库的配置。对于较复杂的环境，可能需要选择“Custom Database（定制数据库）”选项。
4. **Database Identification (数据库标识)**：以 `database_name.domain_name` 的格式输入全局数据库名，然后输入系统标识符 (SID)。SID 的默认值是数据库名称，用于唯一标识与数据库关联的实例。



## 使用 DBCA 创建数据库（续）

5. **Management Options (管理选项)**：使用此页设置数据库，以便可以通过 Oracle Enterprise Manager 管理数据库。选择默认值：“Configure the Database with Enterprise Manager（使用 Enterprise Manager 配置数据库）”。另外，在此页上还可以配置预警通知和每日磁盘备份区域设置。如果 DBCA 检测不到可以通信的 Enterprise Manager Grid Control 代理，则“Register with Grid Control（在 Grid Control 中注册）”选项将灰显。

**注：**为数据库配置 Enterprise Manager 还需要同时配置监听程序（在本课中，监听程序是在安装 Oracle Grid Infrastructure 时配置的）。

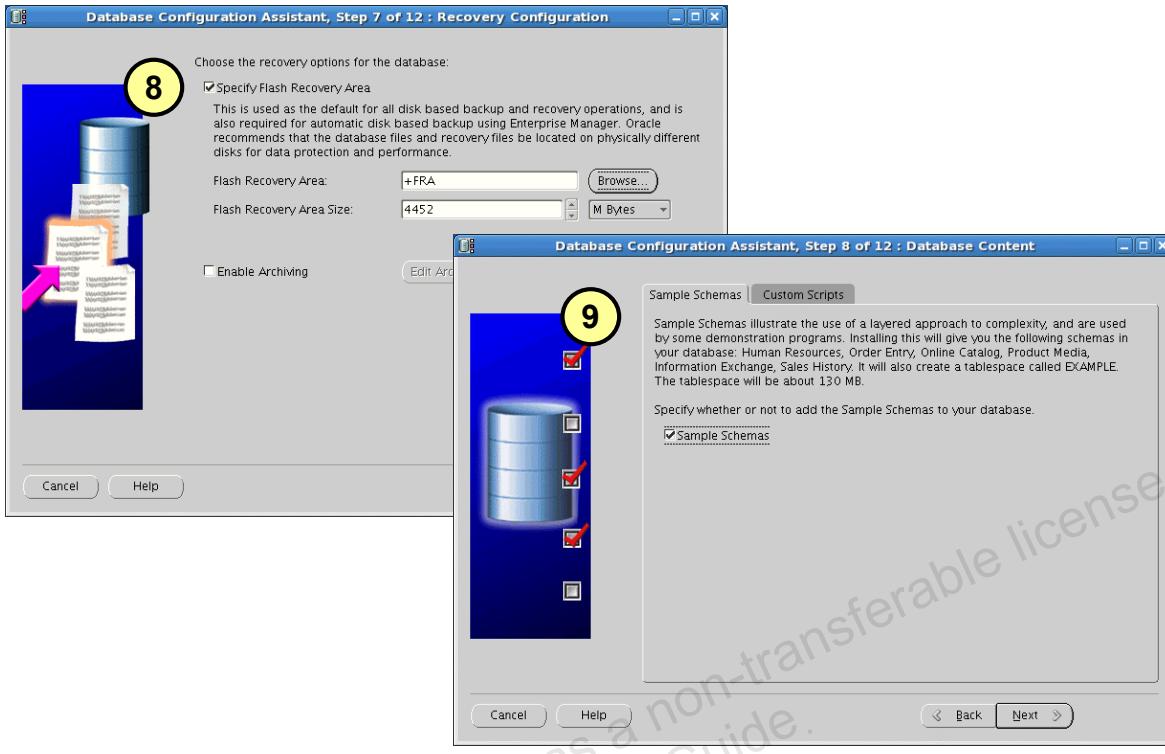
6. **Database Credentials (数据库身份证明)**：使用此页指定管理帐户（如 SYS 和 SYSTEM）的口令。在本课中，使用 `oracle_4U` 作为所有管理帐户的口令。



## 使用 DBCA 创建数据库 (续)

7. **Database File Locations (数据库文件位置)**：指定希望数据库使用的存储机制类型（如“Automatic Storage Management (ASM)（自动存储管理(ASM)）”或“File System（文件系统）”）。对于“Storage Location（存储位置）”，请根据需要进行选择。在本例中，存储机制为 ASM，因此使用 Oracle 管理的文件(OMF)。使用“Browse（浏览）”按钮可以查看可用的磁盘组并选择最适合的磁盘组（幻灯片示例中选择的是 +DATA）。选择磁盘组后，将要求您提供特定于 ASM 的 ASMSNMP 口令。Oracle 管理的文件(OMF)也可与文件系统存储一起使用，这样您就不需要直接管理 Oracle DB 中的操作系统文件。如果需要，您可以单击“Multiplex Redo Logs and Control Files（多路复用重做日志和控制文件）”按钮，为重做日志和控制文件创建其它副本。此页面还提供了一个按钮，用于调整以下文件位置变量：ORACLE\_BASE、ORACLE\_HOME、DB\_NAME、DB\_UNIQUE\_NAME 和 SID。

## 使用 DBCA 创建数据库



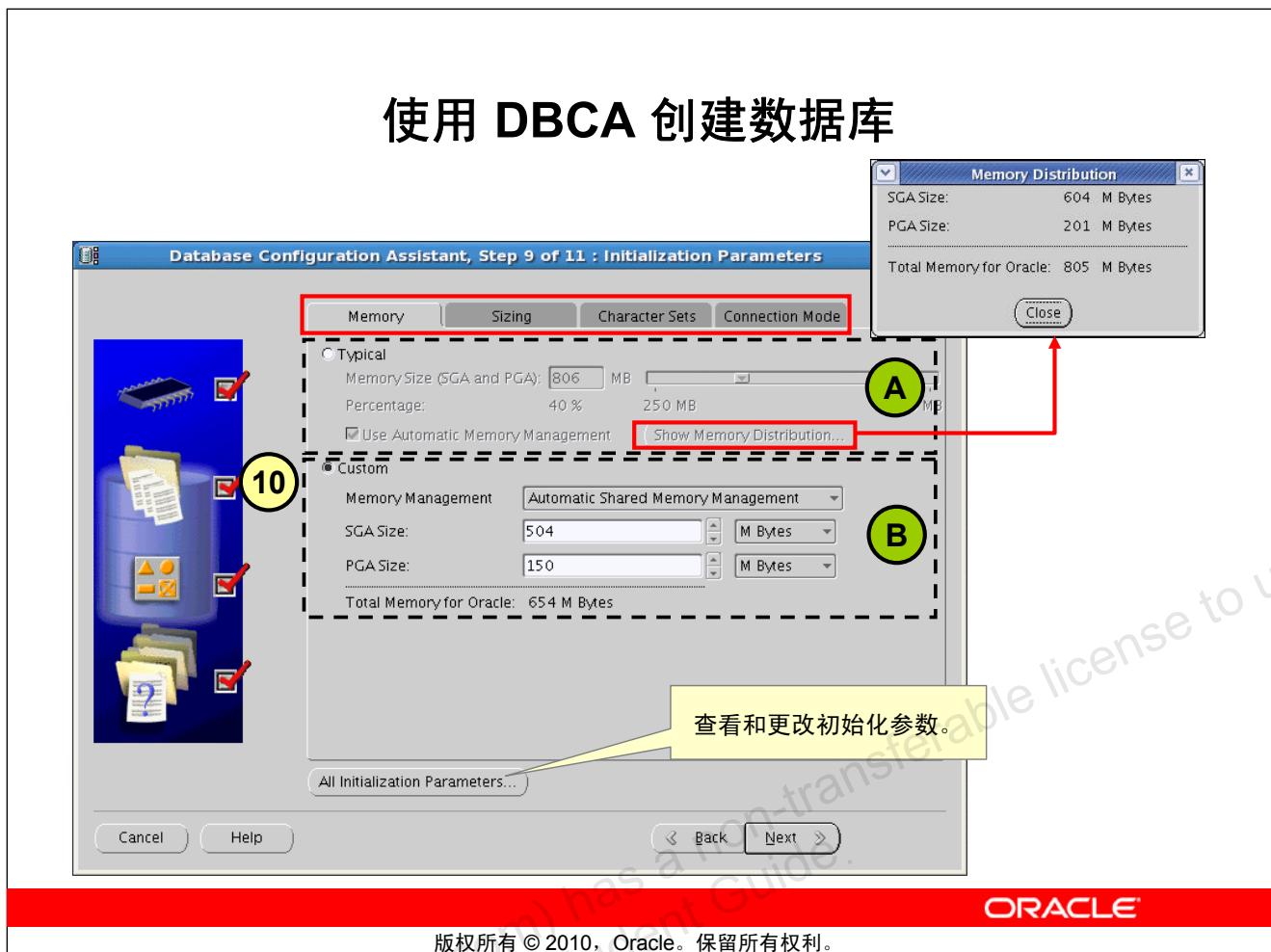
**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 使用 DBCA 创建数据库（续）

8. **Recovery Configuration (恢复配置)**：选择数据库的恢复选项。要配置快速恢复区，请选中“Specify Flash Recovery Area（指定快速恢复区）”选项，然后指定快速恢复区的位置和大小。在显示的示例中，快速恢复区的位置设为使用 +FRA ASM 磁盘组，大小为 4452 MB。建议将快速恢复区的大小设为大于数据库大小的值。归档和与归档相关的参数也可以在此屏幕中进行配置。  
注：快速恢复区的英文名称已从 flash recovery area 改为 fast recovery area，但当前的某些英文版 Oracle 产品可能仍在使用 flash recovery area。
9. **Database Content (数据库内容)**：此页面提供了用于选择组件（例如“Sample Schemas（示例方案）”）的选项和一个选项卡，在该选项卡中可以指定创建数据库后应运行的任何定制脚本。

## 使用 DBCA 创建数据库

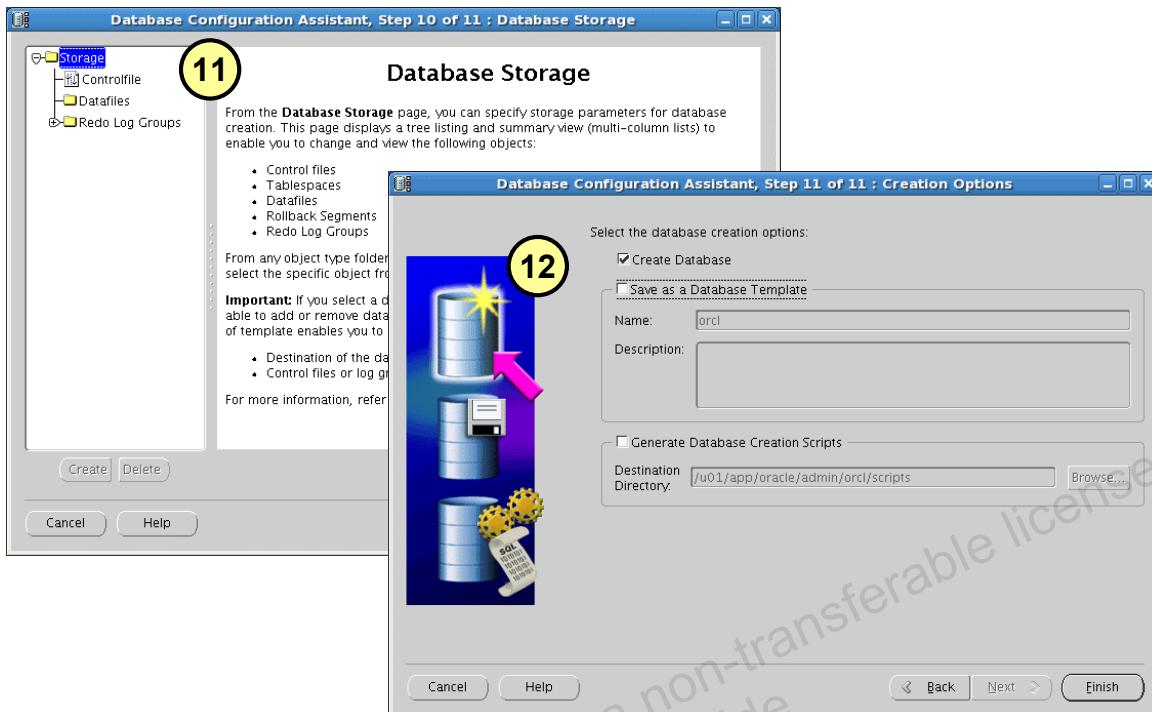


### 使用 DBCA 创建数据库（续）

10. **Initialization Parameters (初始化参数)**： 使用此页中的选项卡可访问用于更改默认数据库设置的页：
  - **Memory (内存)**： 使用此页可设置控制内存使用量的初始化参数。请使用“(A) Typical (典型)”或“(B) Custom (定制)”进行内存分配。
  - **Sizing (调整大小)**： 要指定块大小，请以字节为单位输入大小或接受默认值。
  - **Character Sets (字符集)**： 使用此页可指定数据库的字符集。
  - **最佳做法提示**： Oracle Corporation 建议尽可能使用 Unicode 作为数据库字符集，因为这种字符集可灵活地支持 Web 技术及许多语言。
  - **Connection Mode (连接模式)**： 选择“Dedicated Server Mode (专用服务器模式)”或“Shared Server Mode (共享服务器模式)”。有关详细信息，请参阅“配置 Oracle Network 环境”一课。
  - 单击“All Initialization Parameters (所有初始化参数)”按钮可查看并修改任何初始化参数。

**注：**有些初始化参数在数据库的生命周期内是设定不变的，如 DB\_BLOCK\_SIZE 参数。

## 使用 DBCA 创建数据库



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 使用 DBCA 创建数据库（续）

11. **Database Storage (数据库存储)**：根据需要，查看和修改当前数据库存储设置。如果为数据库选择了预配置模板中的一个，则不能添加或删除控制文件或数据文件。
12. **Create Options (创建选项)**：您可以选择以下选项：现在创建数据库、将数据库定义保存为模板和生成脚本。如果选择所有选项并单击“Finish (完成)”，则 DBCA 首先会保存数据库模板，然后在目标目录中生成脚本，最后创建数据库。

# 创建数据库概要



## 创建数据库概要

13. 单击“Finish（完成）”后会显示确认页，您可以在该页中检查数据库的所有配置，然后再进行创建。单击“OK（确定）”关闭确认页并开始创建数据库。

**注：**为便于参考，可能需要将数据库定义保存为 HTML 文件。

## 口令管理



### 口令管理

14. DBCA 完成后，请记录以下信息以备将来参考：

- 安装日志文件的位置
- 全局数据库名
- 系统标识符 (SID)
- 服务器参数文件名和位置
- Oracle Enterprise Manager URL

单击“Password Management（口令管理）”，取消锁定计划使用的数据库帐户。

取消锁定帐户时，请提供口令。此时未取消锁定的任何帐户都可以在以后根据需要取消锁定。

## 创建数据库设计模板



### 创建数据库设计模板

模板指的是预定义的数据库定义，它可用作创建新数据库的基础。如果在数据库创建过程中未创建模板，可以随时调用 DBCA 并选择“Manage Templates（管理模板）”操作来创建。

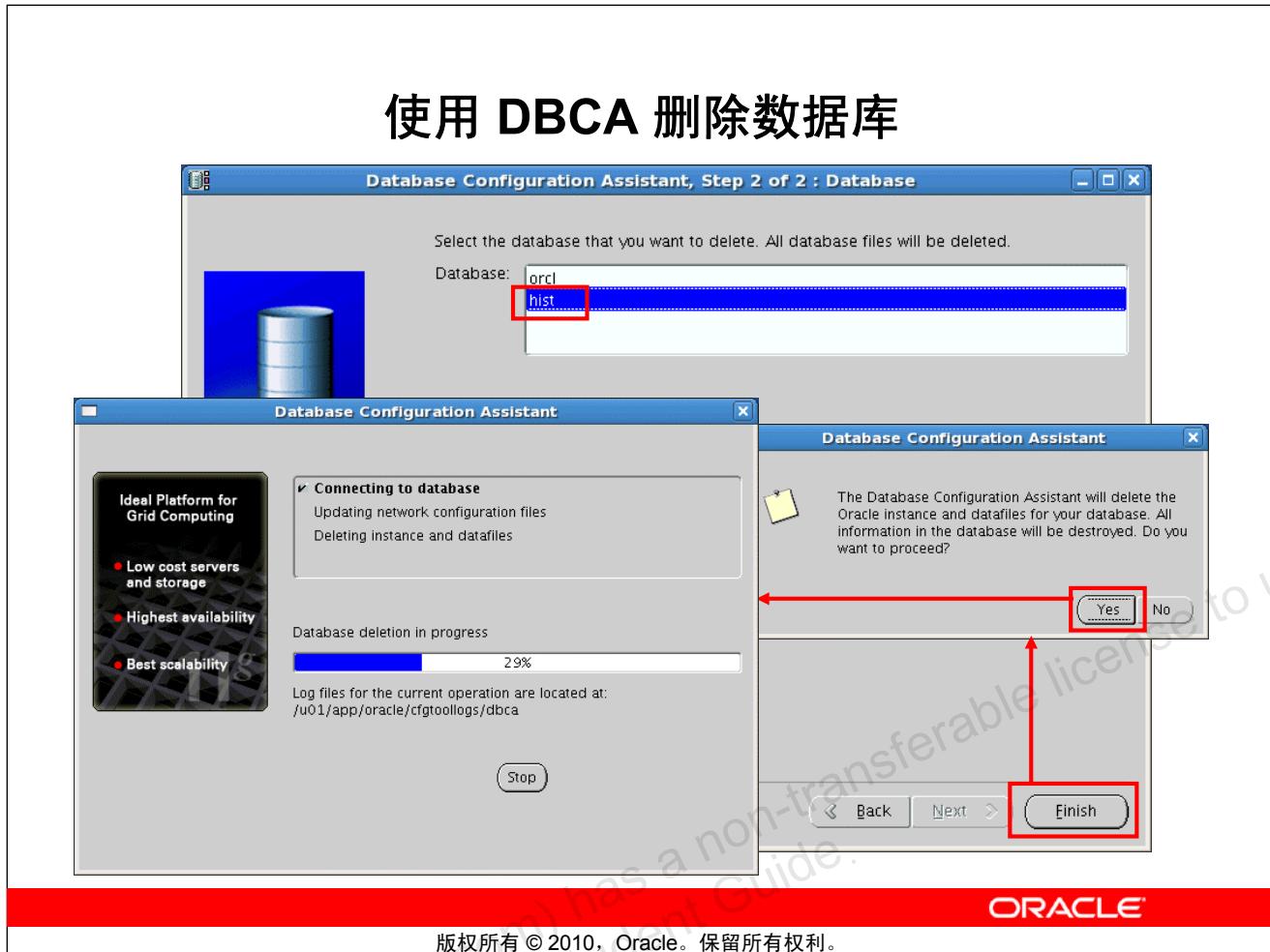
可通过三种方法创建模板：

- 根据现有模板
- 根据现有数据库（仅限结构）
- 根据现有数据库（结构与数据）

DBCA 会指导您完成创建数据库设计模板的各个步骤。

如果不再需要某个特定的模板，可使用 DBCA 的“Template Management（模板管理）”页上的“Delete a database template（删除数据库模板）”选项。

**注：**使用 DBCA 创建新的数据库时，创建的模板将显示在“Database Templates（数据库模板）”列表中。



## 使用 DBCA 删除数据库

在终端窗口中输入 dbca 启动 DBCA，然后单击“Welcome（欢迎使用）”页上的“Next（下一步）”。要删除数据库，请执行以下步骤：

1. 在“Operations（操作）”页中，选择“Delete a Database（删除数据库）”。然后，单击“Next（下一步）”。
2. 选择要删除的数据库（本例中为 hist），然后单击“Finish（完成）”。
3. 单击“Yes（是）”确认删除。
4. 完成删除后，会询问您是否要执行其它操作。相应地做出回应。

**注：**要删除的数据库必须有效且正在运行，这样 DBCA 才能连接到该数据库以确定文件位置信息。

## 使用 DBCA 删除数据库（续）

删除数据库的过程涉及删除其数据文件、重做日志文件、控制文件和初始化参数文件。您可以使用 SQL 语句 DROP DATABASE 手动删除数据库。使用 DROP DATABASE 语句将删除所有控制文件及控制文件中列出的其它所有数据库文件。要成功使用 DROP DATABASE 语句，必须符合以下所有条件：

- 数据库必须已装载且已关闭。
- 数据库必须以独占方式装载（而不是在共享模式下）。
- 数据库必须已经在 RESTRICT 模式下启动。

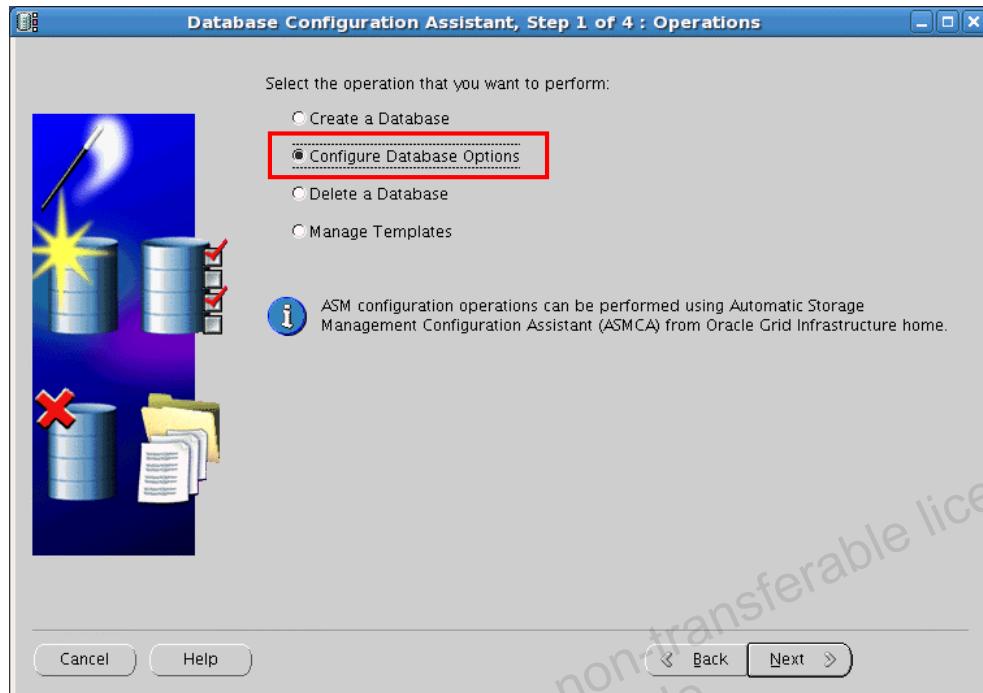
下面是这些语句的一个示例：

```
STARTUP RESTRICT FORCE MOUNT;  
DROP DATABASE;
```

DROP DATABASE 语句对归档日志文件不起作用，对数据库的副本或备份也不起作用。

最好使用 Recovery Manager (RMAN) 来删除这种文件。如果数据库保存在裸盘中，则不会删除实际裸盘专用文件。

## 使用 DBCA 完成其它任务



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 使用 DBCA 完成其它任务

您可以使用 DBCA 配置数据库选件（例如 Oracle Label Security 或 Oracle Database Vault）。安装数据库软件时，并不是默认安装所有选件。因此，需要手动安装这些选件，然后再尝试配置数据库使用这些选件。

### 附注

- 有关 Oracle Label Security 的详细信息，请参阅《Oracle Label Security Administrator's Guide》。
- 有关 Oracle Database Vault 的详细信息，请参阅《Oracle Database Vault Administrator's Guide》。

## 小测验

参数 DB\_BLOCK\_SIZE 在数据库的生命周期内是设定不变的，不能更改。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：1

## 小测验

为了能够使用 DROP DATABASE 命令删除数据库，数据库必须：

1. 打开并处于 RESTRICT 模式下
2. 在 RESTRICT 模式下以独占方式装载
3. 使用“Immediate（立即）”选项关闭

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2

## 小结

在本课中，您应该已经学会：

- 使用 Database Configuration Assistant (DBCA) 创建数据库
- 使用 DBCA 生成数据库创建脚本
- 使用 DBCA 管理数据库设计模板
- 使用 DBCA 执行其它任务

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 3 概览：使用 DBCA

本练习包含以下主题：

- 使用 DBCA 创建 ORCL 数据库
- 取消锁定 HR 方案

**注：完成数据库创建和取消锁定 HR 方案对后面的所有练习来说至关重要。**

- 使用 DBCA 创建 ORCL 数据库设计模板
- 使用 DBCA 创建数据库创建脚本

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



## 管理数据库实例

4

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 启动和停止 Oracle DB 和组件
- 使用 Oracle Enterprise Manager
- 使用 SQL\*Plus 访问数据库
- 修改数据库初始化参数
- 描述数据库启动阶段
- 描述数据库关闭选项
- 查看预警日志
- 访问动态性能视图

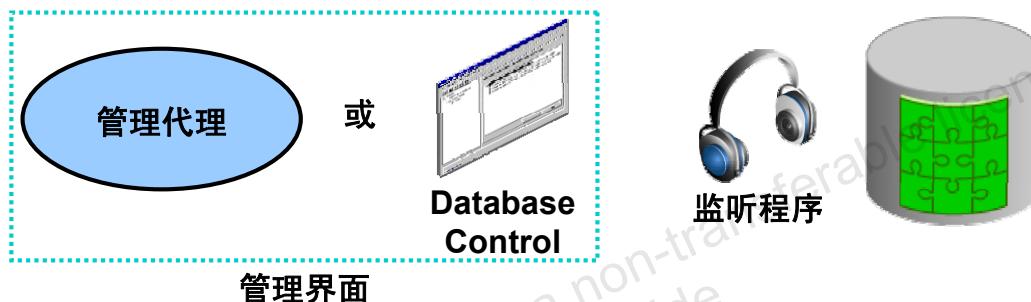
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 管理框架

Oracle Database 11g 发行版 2 管理框架组件包括：

- 数据库实例
- 监听程序
- 管理界面：
  - Database Control
  - 管理代理（使用 Grid Control 时）



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 管理框架

在 Oracle DB 管理框架中包括以下三个主要组件：

- 正在被管理的数据库实例。
- 用于连接到数据库的监听程序。
- 管理界面。可以是在数据库服务器所运行的节点上运行的管理代理（通过数据库服务器将管理代理连接到 Oracle Enterprise Manager Grid Control），也可以是独立的 Oracle Enterprise Manager Database Control。该界面也称为 *Database Console*。

必须启动每一个组件后才能使用相应组件提供的服务，另外在关闭托管 Oracle DB 的服务器时必须彻底关闭每个组件。

## 启动和停止 Database Control

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/db_home1
is /u01/app/oracle
$ emctl start dbconsole
Oracle Enterprise Manager 11g Database Control Release 11.2.0.1.0
Copyright (c) 1996, 2009 Oracle Corporation. All rights reserved.
http://host01.example.com:1158/em/console/aboutApplication
Starting Oracle Enterprise Manager 11g Database Control .....started.
-----
Logs are generated in directory
/u01/app/oracle/product/11.2.0/db_home1/host01.example.com_orcl/sysman/
log
```

```
$ emctl stop dbconsole
Oracle Enterprise Manager 11g Database Control Release 11.2.0.1.0
Copyright (c) 1996, 2009 Oracle Corporation. All rights reserved.
https://host01.example.com:1158/em/console/aboutApplication
Stopping Oracle Enterprise Manager 11g Database Control ...
... Stopped.
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 启动和停止 Database Control

Oracle DB 为没有连接到 Grid Control 框架的数据库提供一个称为 *Database Control* 的独立管理控制台。若用 Database Control 管理，则每个数据库都将分别安装一个 Database Control；任何一个 Database Control 都只能管理一个数据库。使用 Database Control 之前，请确保 dbconsole 进程已启动。

用于启动 dbconsole 进程的命令：

```
emctl start dbconsole
```

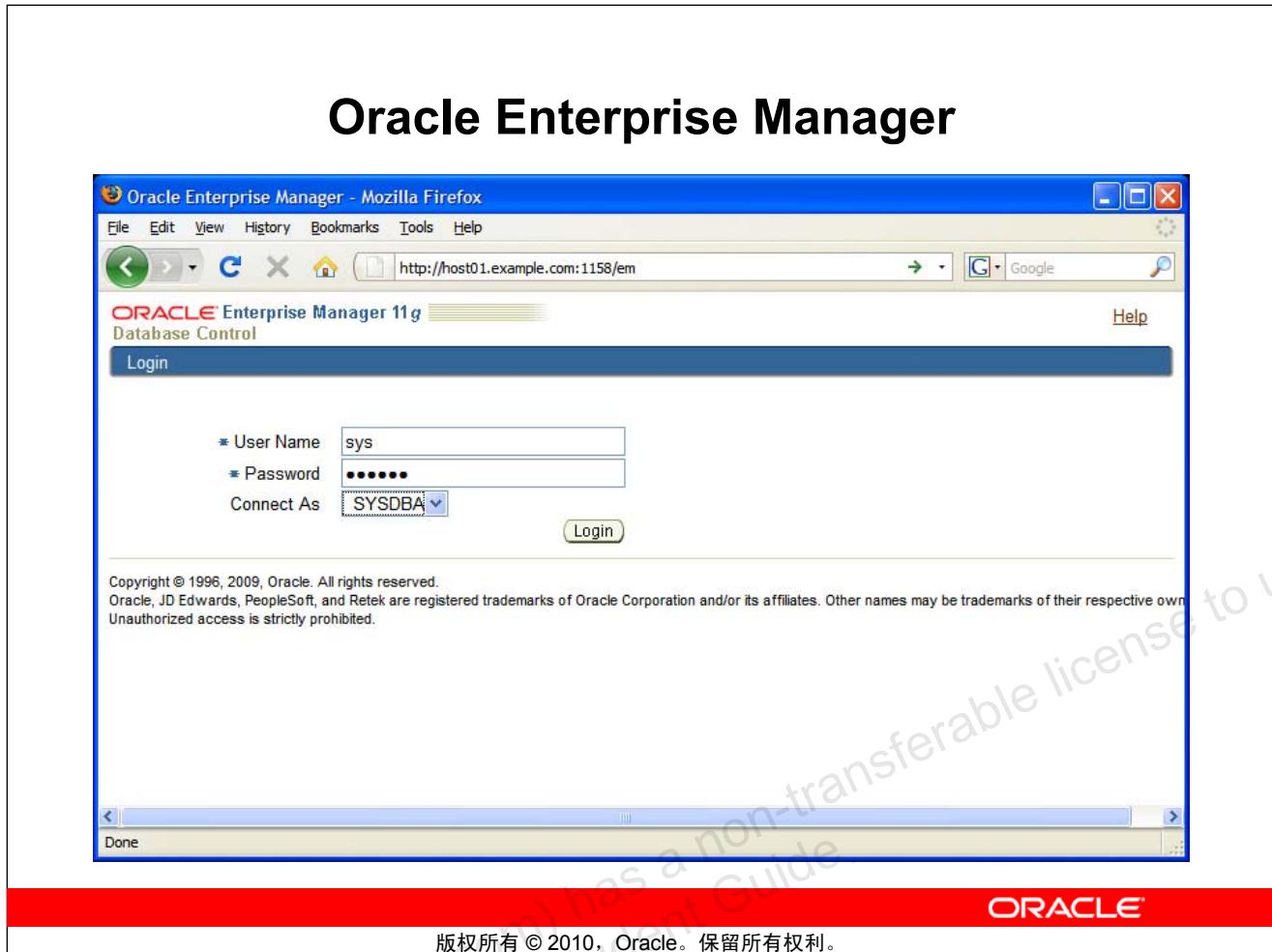
用于停止 dbconsole 进程的命令：

```
emctl stop dbconsole
```

用于查看 dbconsole 进程状态的命令：

```
emctl status dbconsole
```

**注：**如果 \$ORACLE\_HOME/bin 目录不在操作系统 (OS) 路径中，则可能需要导航到此目录。如果安装了 Oracle Grid Infrastructure，则会有两个 \$ORACLE\_HOME 位置，且二者都含有 emctl 实用程序。调用 emctl 实用程序时，应始终使用 Oracle DB 的 \$ORACLE\_HOME，而不是 Oracle Grid Infrastructure 的 \$ORACLE\_HOME。Database Control 使用一个服务器端代理进程。dbconsole 进程启动或停止时，该代理进程会相应地自动启动或停止。



## Oracle Enterprise Manager

安装 Oracle DB 软件时，Oracle Universal Installer (OUI) 还会安装 Oracle Enterprise Manager (Enterprise Manager)。Enterprise Manager 中基于 Web 的 Database Control 部件是作为管理 Oracle DB 的主要工具。Enterprise Manager 提供了一个图形界面，使用该界面几乎可完成数据库管理员 (DBA) 需要完成的所有任务。使用 Enterprise Manager 可完成的任务很多，例如，查看预警概要与性能图、创建和修改对象以及执行备份与恢复等等。在大多数情况下，单击 Enterprise Manager 中的链接可以找到有关页面内容的更多具体信息。

**注：**在 Oracle Database 11g 发行版 2 中，用于访问 Enterprise Manager 的 URL 使用 HTTPS (而不是 HTTP) 协议，以启用安全连接。要访问 Enterprise Manager dbconsole，必须采用如下格式输入 URL：

```
https://machine_name:port/em
```

对于在计算机上创建的第一个数据库来说，访问 Enterprise Manager Database Control 的默认端口号为 1158。您可以使用其它编号，尤其是在同一主机上存在多个数据库的情况下。要确定端口号，请查看 portlist.ini 文件。portlist.ini 文件中列出了一些 Oracle DB 应用程序的端口，该文件位于 \$ORACLE\_HOME/install 目录。

## Oracle Enterprise Manager (续)

输入 Enterprise Manager 的 URL 时，所显示的内容取决于数据库的状态：

- 如果数据库已启动，Enterprise Manager 会显示 Database Control 的登录页。请使用授权访问 Database Control 的用户名登录数据库。初始的用户名是 SYS、SYSMAN 或 SYSTEM。请使用在数据库安装期间为帐户指定的口令。在“Connect As (连接身份)”选项中，选择具有特殊数据库管理权限的 Normal 或 SYSDBA 登录到数据库。
- 如果数据库已关闭，Enterprise Manager 会显示“Startup/Shutdown and Perform Recovery (启动/关闭并执行恢复)”页。如果是这样，请单击“Startup/Shutdown (启动/关闭)”按钮。然后，系统会提示您输入主机和目标数据库的登录用户名和口令，这些是必须输入的内容。

注：如果启动 Enterprise Manager 时出现了问题，请确保启动了监听程序。

## “Database（数据库）”主页

The screenshot shows the Oracle Enterprise Manager 11g Database Control interface. At the top, it says "ORACLE Enterprise Manager 11g Database Control" and "Logged in As SYS". The main title is "Database Instance: orcl.example.com". Below the title, there are tabs: Home, Performance, Availability, Server, Schema, Data Movement, Software and Support, and Properties (which is highlighted with a red box). A message at the top indicates the page was refreshed on Jun 18, 2009, at 11:46:00 PM GMT+07:00. There are four main sections: General, Host CPU, Active Sessions, and SQL Response Time. The General section shows the instance status as "Up" since Jun 18, 2009, at 5:31:03 AM GMT+07:00. It also lists the instance name (orcl), version (11.2.0.1.0), host (edrsr25p1.us.oracle.com), listener (LISTENER\_edrsr25p1.us.oracle.com), and ASM (ASM\_edrsr25p1.us.oracle.com). The Host CPU section has a chart showing CPU usage with "orcl" as the active session. The Active Sessions section shows one session. The SQL Response Time section shows a chart for the latest collection. The Diagnostic Summary section shows ADDM findings (2), a period from Jun 18, 2009, 10:00:40 PM GMT+07:00, and no ORA errors. The Space Summary section shows database size (1.448 GB), problem tablespaces (0), segment advisor recommendations (0), policy violations (0), and dump area used (85%). The High Availability section shows console (Enabled), oracle restart (14), instance recovery time (n/a), last backup (n/a), usable flash recovery area (95.96%), and flashback database logging (Disabled). The bottom right corner features the ORACLE logo.

## “Database（数据库）”主页

“Database（数据库）”主页显示一组描述数据库总体运行状况的度量数据来显示数据库的当前状态。借助属性页（又称为“选项卡”），您可以访问“Performance（性能）”、“Availability（可用性）”、“Server（服务器）”、“Schema（方案）”、“Data Movement（数据移动）”和“Software and Support（软件和支持）”页，从而管理数据库。

在数据库主页上，您可以查看数据库实例的以下性能和状态信息：

- 实例名称、数据库版本、Oracle 主目录位置、介质恢复选项和其它相关的实例数据
- 当前实例可用性
- 未处理预警
- 与会话和 SQL 相关的性能信息
- 主要空间使用情况度量
- 提供更具体详细信息的细化链接（例如，LISTENER\_<host\_name>）

## 其它 Oracle 工具

组件  
 > **SQL\*Plus**  
 初始化参数  
 DB 启动  
 DB 关闭  
 预警日志  
 性能视图

- SQL\*Plus 为数据库操作提供了另一个界面，以便您：
  - 执行数据库管理操作
  - 通过执行 SQL 命令在数据库中查询、插入、更新或删除数据
- SQL Developer：
  - 是一个用于访问 Oracle DB 实例的图形用户界面
  - 支持以 SQL 和 PL/SQL 进行开发
  - 在 Oracle DB 的默认安装中提供



版权所有 © 2010, Oracle。保留所有权利。

### 其它 Oracle 工具

除 Enterprise Manager 外，您还可以使用 SQL\*Plus 和 SQL Developer 发出 SQL 语句。使用这些工具可执行多项数据库管理操作，还可在数据库中选择、插入、更新或删除数据。

SQL\*Plus 是一个命令行程序，用于向 Oracle DB 提交 SQL 和 PL/SQL 语句。提交语句时可以采用交互方式，也可以将语句作为 SQL\*Plus 脚本提交。SQL\*Plus 随数据库一起安装，位于 \$ORACLE\_HOME/bin 目录中。

可以从命令行或 Windows 客户机上的“开始”菜单中启动 SQL\*Plus。

SQL Developer 是一个用于访问 Oracle DB 实例的图形用户界面。SQL Developer 支持以 SQL 和 PL/SQL 语言进行开发。默认 Oracle DB 安装中提供该工具。

使用 SQL Developer 可以浏览数据库对象、运行 SQL 语句和 SQL 脚本以及编辑和调试 PL/SQL 语句。您还可以运行该工具提供的任意报告，也可以创建和保存您自己的报告。

注：本课程使用 Enterprise Manager 和 SQL\*Plus。

## 使用 SQL\*Plus

**SQL\*Plus:**

- 是一个命令行工具
- 可以通过交互方式或者在批处理模式下使用

```
$ sqlplus hr

SQL*Plus: Release 11.2.0.1.0 - Production on Thu Jun 18 05:04:49 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.
Enter Password: *****

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

SQL> select last_name from employees;
LAST_NAME
-----
Abel
Ande
...

```



版权所有 © 2010, Oracle。保留所有权利。

### 使用 SQL\*Plus

通过使用 SQL\*Plus 的命令行界面执行 SQL\*Plus、SQL 和 PL/SQL 命令，可以：

- 输入、编辑、运行、存储、检索和保存 SQL 命令和 PL/SQL 块
- 格式化、计算、存储和打印查询结果
- 列出任何表的列定义
- 向最终用户发送消息或接受来自最终用户的响应
- 执行数据库管理

要启动 SQL\*Plus，请执行以下操作：

1. 打开终端窗口。
2. 在命令行提示符下，按以下格式输入 SQL\*Plus 命令：  
`$ sqlplus <userid>/<pwd> or /nolog`
3. 如果使用了 NOLOG 选项，则必须输入 CONNECT，后跟要作为连接身份的用户名。  
`SQL> connect <username>`
4. 出现提示后，输入该用户的口令。SQL\*Plus 启动后会连接到默认数据库。

## 从 Shell 脚本调用 SQL\*Plus

```
$ ./batch_sqlplus.sh ←
SQL*Plus: Release 11.2.0.1.0 - Production on Thu Jun 18 05:10:19 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options

SQL>
  COUNT(*)
-----
      107
SQL>
107 rows updated.
SQL>
Commit complete.
SQL> Disconnected from Oracle Database 11g Enterprise Edition Release
11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options
$
```

输出

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 从 Shell 脚本调用 SQL\*Plus

通过调用 sqlplus 和使用操作系统脚本语法传递参数，可从 shell 脚本或 BAT 文件中调用 SQL\*Plus。

在本例中，先执行 SELECT、UPDATE 和 COMMIT 语句，之后 SQL\*Plus 将控制权返回给操作系统。

## 从 SQL\*Plus 调用 SQL 脚本

**script.sql**

```
select * from departments where location_id = 1400;
quit
```

↓  
输出

```
$ sqlplus hr/hr @script.sql
```

```
SQL*Plus: Release 11.2.0.1.0 - Production on Thu Jun 18 05:13:42 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.
```

```
Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options
```

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
60	IT	103	1400

```
Disconnected from Oracle Database 11g Enterprise Edition Release
11.2.0.1.0 - Production
With the Partitioning, Automatic Storage Management, OLAP, Data Mining
and Real Application Testing options
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

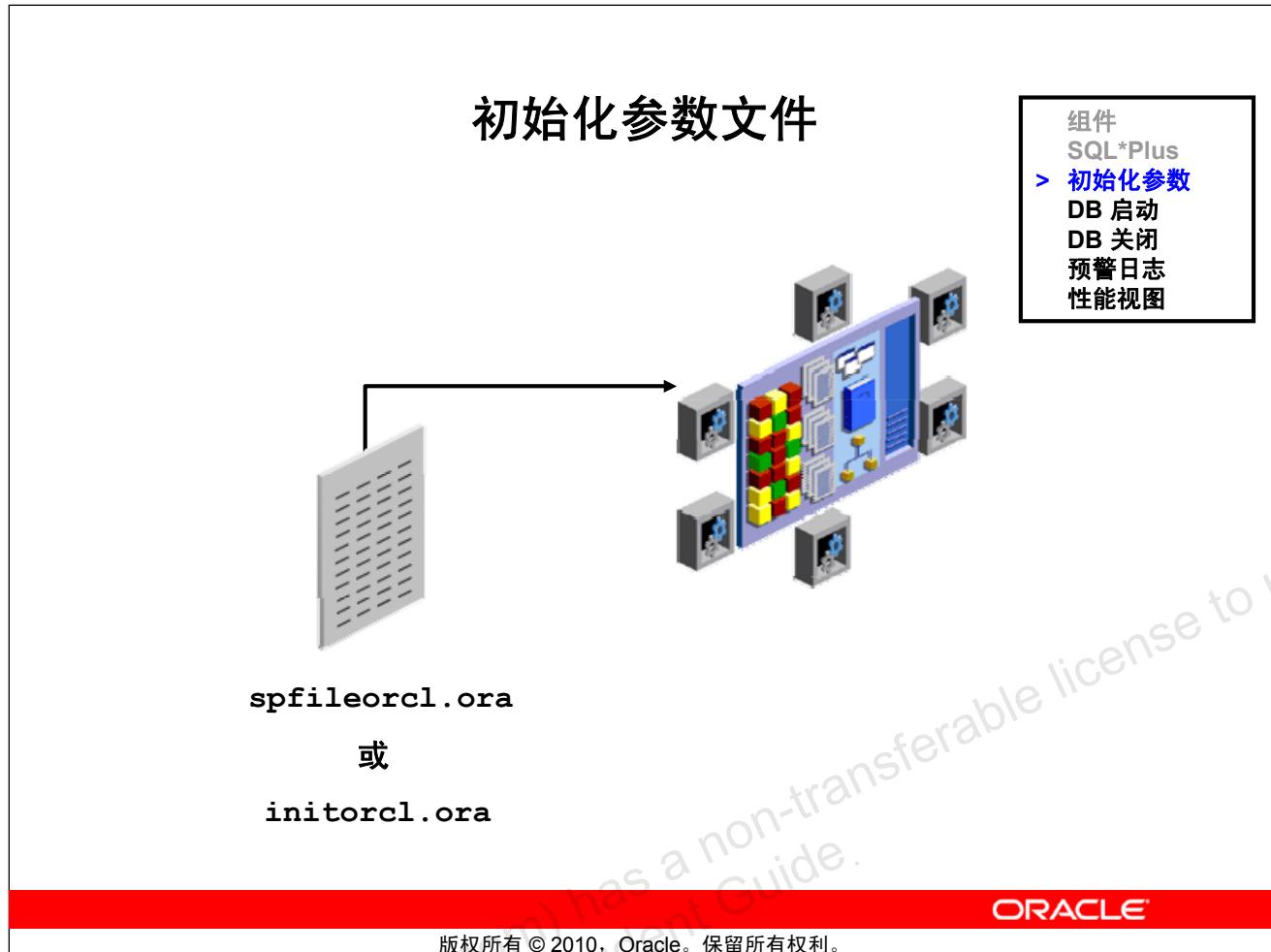
## 从 SQL\*Plus 调用 SQL 脚本

您可以从 SQL\*Plus 中调用现有的 SQL 脚本文件。可在首次调用 SQL\*Plus 时通过命令行完成此操作，如幻灯片所示。也可以使用“@”运算符，从 SQL\*Plus 会话中完成此操作。例如，输入以下内容可从已建立的 SQL\*Plus 会话中运行脚本：

```
SQL> @script.sql
```

**注：**脚本文件的默认文件扩展名为 .sql。使用 Save 命令从 SQL\*Plus 保存脚本时，会自动提供该扩展名。在执行时即使不提供该扩展名，也可以执行具有该扩展名的脚本，如下面的示例所示：

```
SQL> @script
```



## 初始化参数文件

启动实例时会读取初始化参数文件。参数文件有两种类型。

- **服务器参数文件 (SPFILE):** 这是首选的初始化参数文件类型。这是一个可由数据库服务器写入和读取的二进制文件，不得手动进行编辑。此文件驻留在执行 Oracle 实例的服务器上，而且永远存在，不受数据库关闭和启动的影响。此文件的默认名称为 `spfile<SID>.ora`，启动时会自动搜索此文件。
- **文本初始化参数文件:** 这类初始化参数文件可由数据库服务器读取，但不由其写入。必须使用文本编辑器手动设置和更改初始化参数设置，这些设置是永久的，不受数据库关闭或启动的影响。此文件的默认名称为 `init<SID>.ora`，如果启动时未找到 SPFILE，则自动搜索此文件。

建议创建 SPFILE，从而以动态方式维护初始化参数。

注：在 Linux 上，Oracle DB 会在 `$ORACLE_HOME/dbs` 目录中搜索初始化文件。采用 ASM 时，SPFILE 通常位于 ASM 磁盘组中。在这种情况下，`$ORACLE_HOME/dbs` 目录中应当有一个 `init<SID>.ora` 文件，该文件标识了 SPFILE 的位置。

## 初始化参数文件（续）

### 初始化参数值的类型

Oracle DB 服务器有下列类型的初始化参数值：

- Boolean
- String
- Integer
- 参数文件
- 保留
- Big Integer

### 派生参数值

一些初始化参数是派生的，也就是说它们的值是通过计算其它参数的值得来的。通常，您不应变更派生参数的值。但如果进行了变更，您指定的值将覆盖计算得出的值。

例如，SESSIONS 参数的默认值由 PROCESSES 参数的值派生。如果 PROCESSES 的值发生更改，则 SESSIONS 的默认值也会发生更改，除非您指定一个值来覆盖它。

### 与操作系统相关的参数值

一些初始化参数的有效值或值范围与主机操作系统相关。例如，

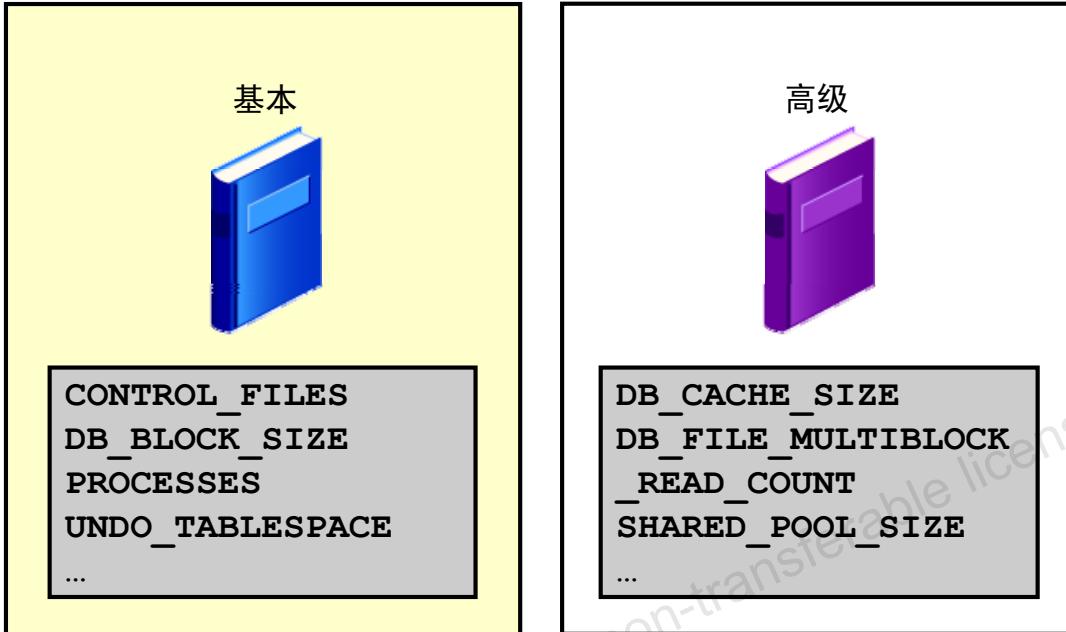
DB\_FILE\_MULTIBLOCK\_READ\_COUNT 参数指定在一次顺序扫描期间在一个 I/O 操作中读取的最大块数；此参数与平台相关。这些块的大小（由 DB\_BLOCK\_SIZE 设置）的默认值与操作系统相关。

### 设置参数值

初始化参数最有可能改进系统性能。一些参数设置容量限制，但不影响性能。例如，当 OPEN\_CURSORS 的值为 10 时，用户进程如果尝试打开第 11 个游标，将收到错误消息。有些参数会影响性能，但不施加绝对限制。例如，减小 OPEN\_CURSORS 的值可能会降低性能，但不会阻止工作。

增大参数值也许可以改善系统性能，但增大大多数参数也会增加系统全局区 (SGA) 大小。较大的 SGA 可以将数据库性能提升至某个点。在虚拟内存操作系统中，如果 SGA 是交换进/出内存的，则太大的 SGA 会降低性能。设置控制虚拟内存工作区域的操作系统参数时，应考虑 SGA 大小。操作系统配置也会限制 SGA 的最大大小。

## 简化初始化参数



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 简化初始化参数

初始化参数有两种类型：基本初始化参数和高级初始化参数。

大多数情况下，只需设置和优化 30 个基本参数便可获得合理的数据库性能。只有极少数情况下，才需要修改高级参数来获得最优性能。高级参数大约有 314 个。

基本参数是指为了让数据库以良好性能运行而可能要设置的参数。其它所有参数都被视为高级参数。

基本参数示例：

- 确定全局数据库名称：DB\_NAME 和 DB\_DOMAIN
- 指定快速恢复区和大小：DB\_RECOVERY\_FILE\_DEST 和 DB\_RECOVERY\_FILE\_DEST\_SIZE
- 指定所有 SGA 组件的总大小：SGA\_TARGET
- 指定还原空间管理表空间的方法：UNDO\_TABLESPACE
- COMPATIBLE 初始化参数和不可逆兼容性

注：以下页面中列出了一些初始化参数。有关完整的列表，请参阅《Oracle Database Reference》。

## 初始化参数：示例

参数	指定
CONTROL_FILES	一个或多个控制文件名
DB_FILES	最大数据库文件数
PROCESSES	可以同时连接的最大操作系统用户进程数
DB_BLOCK_SIZE	用于所有表空间的数据库标准块大小
DB_CACHE_SIZE	标准块缓冲区高速缓存的大小

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 初始化参数：示例

**CONTROL\_FILES 参数：**指定一个或多个控制文件名。Oracle 强烈建议您多路复用控制文件并为其创建镜像。值范围：从一个到八个文件名（带路径名）。默认值：与操作系统相关。

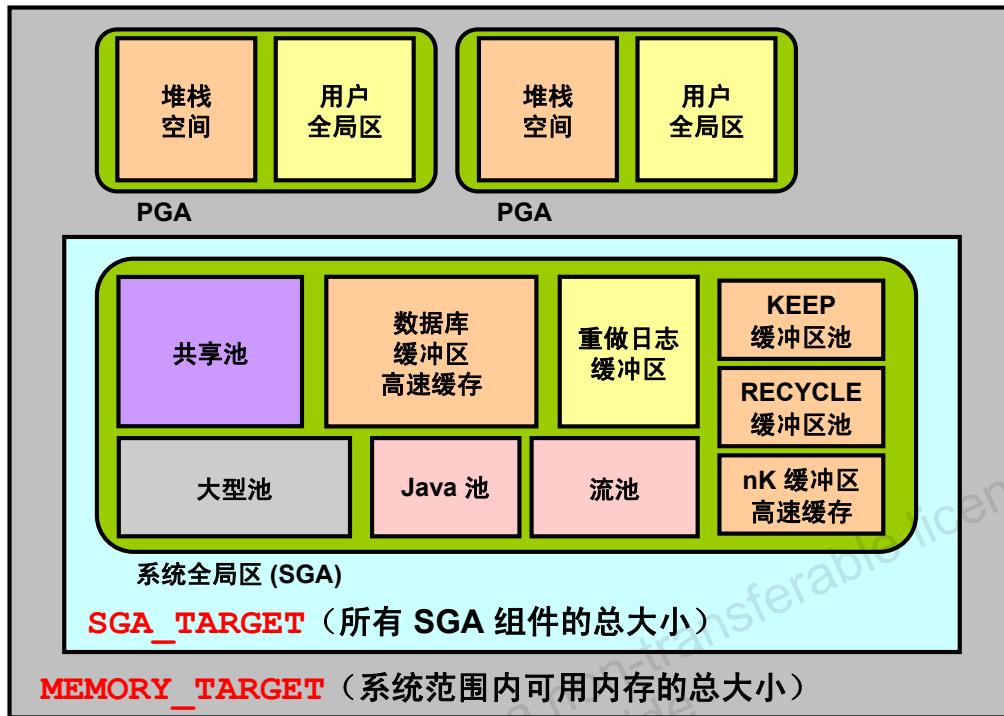
**DB\_FILES 参数：**指定可为此数据库打开的最大数据库文件数。值范围：与操作系统相关。默认值：200。

**PROCESSES 参数：**指定可同时连接到 Oracle 服务器的最大操作系统用户进程数。此值应允许所有后台进程和用户进程连接到 Oracle 服务器。值范围：从 6 到一个与操作系统相关的值。默认值：100。

**DB\_BLOCK\_SIZE 参数：**指定一个 Oracle DB 块的大小（字节）。此值是在创建数据库时设置的，此后不可更改。此值指定了数据库的标准块大小。所有表空间默认情况下将使用该大小。值范围：2048 到 32768（与操作系统相关）。默认值：8192。

**DB\_CACHE\_SIZE 参数：**指定标准块缓冲区高速缓存的大小。值范围：至少为 16 MB。默认值：如果设置了 SGA\_TARGET，则为 0；否则为 48 MB 与 (4 MB \* cpu\_count) 中的较大者。

## 初始化参数：示例



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 初始化参数：示例（续）

**SGA\_TARGET** 指定所有 SGA 组件的总大小。如果指定了 **SGA\_TARGET**，则会自动调整以下内存池的大小：

- 缓冲区高速缓存 (**DB\_CACHE\_SIZE**)
- 共享池 (**SHARED\_POOL\_SIZE**)
- 大型池 (**LARGE\_POOL\_SIZE**)
- Java 池 (**JAVA\_POOL\_SIZE**)
- 流池 (**STREAMS\_POOL\_SIZE**)

如果将这些自动调整的内存池设置为非零值，则自动共享内存管理 (ASMM) 功能将这些值作为下限值。如果某个应用程序组件至少需要一定数量的内存才能正常使用，您应当设置下限值。

以下池是手动调整大小的组件，不受 ASMM 的影响：

- 日志缓冲区
- 其它缓冲区高速缓存（如 **KEEP** 和 **RECYCLE**）和其它块大小
- 固定 SGA 和其它内部分配

启用 ASMM 时，会从 **SGA\_TARGET** 的总可用内存中减去分配给上述这些池的内存。

## 初始化参数：示例（续）

注：MMON 进程会计算自动调整的内存池的值，以支持 ASMM。

MEMORY\_TARGET 指定 Oracle 系统范围内可用的内存。数据库将内存调整为 MEMORY\_TARGET 值，并根据需要减小或增加 SGA 和 PGA。

在基于文本的初始化参数文件中，如果省略了 MEMORY\_MAX\_TARGET 而包含了 MEMORY\_TARGET 的值，则数据库会自动将 MEMORY\_MAX\_TARGET 设置为 MEMORY\_TARGET 的值。如果省略了 MEMORY\_TARGET 行而包含了 MEMORY\_MAX\_TARGET 的值，则 MEMORY\_TARGET 参数默认为零。启动后，可以将 MEMORY\_TARGET 动态地更改为非零值，但该值不能超过 MEMORY\_MAX\_TARGET 的值。可以使用 ALTER SYSTEM 命令修改 MEMORY\_TARGET 参数。值的范围是从 152 MB 到 MEMORY\_MAX\_TARGET。

## 初始化参数：示例

参数	指定
PGA_AGGREGATE_TARGET	分配给所有服务器进程的 PGA 内存量
SHARED_POOL_SIZE	共享池大小（字节）
UNDO_MANAGEMENT	要使用的还原空间管理模式

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 初始化参数：示例（续）

**PGA\_AGGREGATE\_TARGET 参数：**指定分配给此实例所挂接的所有服务器进程的程序全局区 (PGA) 内存量。这些内存不驻留在系统全局区 (SGA) 中。数据库将此参数用作要使用的目标 PGA 内存量。设置此参数时，请从系统上可供 Oracle 实例使用的总内存中减去 SGA。值范围由整数加上字母 *K*、*M* 或 *G*（分别以 KB、MB 或 GB 为单位指定此限制）构成。最小值是 10 MB，最大值是 (4096 GB – 1)。默认值是 10 MB 或 SGA 大小的 20%（取两者中较大的值）。

**SHARED\_POOL\_SIZE 参数：**指定共享池的大小（字节）。共享池包含诸如共享游标、存储过程、控制结构和并行执行消息缓冲区等对象。值范围：与操作系统相关。默认值：如果设置了 SGA\_TARGET，则为 0；否则为 128 MB（64 位）或 48 MB（32 位）。

**UNDO\_MANAGEMENT 参数：**指定系统应使用的还原空间管理模式。如果设置为 AUTO，则在自动还原管理 (AUM) 模式下启动实例。否则，在回退还原 (RBU) 模式下启动实例。在 RBU 模式下，还原空间在形式上被分配为回退段。在 AUM 模式下，还原空间在形式上被分配为还原表空间。值范围：AUTO 或 MANUAL。如果启动第一个实例时省略了 UNDO\_MANAGEMENT 参数，则使用默认值 AUTO。

## 使用 SQL\*Plus 查看参数

```

SQL> SELECT name , value FROM V$PARAMETER;
NAME          VALUE
-----
lock_name_space      2
processes           150
sessions            247
timed_statistics    TRUE
timed_os_statistics 0
...

SQL>SHOW PARAMETER SHARED_POOL_SIZE
NAME          TYPE          VALUE
-----
shared_pool_size      big integer 0

SQL> show parameter para
NAME          TYPE          VALUE
-----
fast_start_parallel_rollback   string        LOW
parallel_adaptive_multi_user  boolean       TRUE
parallel_automatic_tuning     boolean       FALSE
parallel_execution_message_size integer     16384
parallel_instance_group        string
...

```

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 使用 SQL\*Plus 查看参数

幻灯片中显示的是使用 SQL\*Plus 查看参数的示例。可以查询 V\$PARAMETER 数据字典视图来查找各个参数的值。V\$PARAMETER 会显示当前会话中的当前参数值。还可以使用可跟任意字符串的 SHOW PARAMETER 命令来查看包含该字符串的参数。

以下示例中的查询请求的是参数的名称和值，该示例使用 WHERE 子句来指定特定参数名称：

```

SQL> SELECT name, value FROM V$PARAMETER WHERE name LIKE
'%pool%';
NAME          VALUE
-----
shared_pool_size      0
large_pool_size       0
java_pool_size        0
streams_pool_size     0
shared_pool_reserved_size 8808038
buffer_pool_keep
...

```

9 rows selected.

## 使用 SQL\*Plus 查看参数（续）

视图的说明：

SQL> desc V\$parameter	Name	Null?	Type
	NUM		NUMBER
	NAME		VARCHAR2 (80)
	TYPE		NUMBER
	VALUE		VARCHAR2 (4000)
	DISPLAY_VALUE		VARCHAR2 (4000)
	ISDEFAULT		VARCHAR2 (9)
	ISSES_MODIFIABLE		VARCHAR2 (5)
	ISSYS_MODIFIABLE		VARCHAR2 (9)
	ISINSTANCE_MODIFIABLE		VARCHAR2 (5)
	ISMODIFIED		VARCHAR2 (10)
	ISADJUSTED		VARCHAR2 (5)
	ISDEPRECATED		VARCHAR2 (5)
	ISBASIC		VARCHAR2 (5)
	DESCRIPTION		VARCHAR2 (255)
	UPDATE_COMMENT		VARCHAR2 (255)
	HASH		NUMBER

第二个示例显示使用 SQL\*Plus 的 SHOW PARAMETER 命令查看参数设置。您还可以使用该命令查找包含某个文本字符串的所有参数。例如，可以使用以下命令查找包括字符串 db 的所有参数名：

SQL> show parameter db	NAME	TYPE	VALUE
...			
db_8k_cache_size		big integer	0
db_block_buffers		integer	0
db_block_checking		string	FALSE
db_block_checksum		string	TYPICAL
db_block_size		integer	8192
db_cache_advice		string	ON
db_cache_size		big integer	0
...			

其它包含参数相关信息的视图

- V\$SPPARAMETER：显示有关服务器参数文件内容的信息。如果未使用服务器参数文件来启动实例，则该视图每一行的 ISSPECIFIED 列中的值为 FALSE。
- V\$PARAMETER2：显示有关当前对于会话有效的初始化参数的信息，每个参数值显示为视图中的一行。新会话将从 V\$SYSTEM\_PARAMETER2 视图中显示的实例范围的值继承参数值。
- V\$SYSTEM\_PARAMETER：显示有关当前对于实例有效的初始化参数的信息。

## 更改初始化参数值

- 静态参数:
  - 只能在参数文件中更改
  - 必须重新启动实例才能生效
  - 总共约 110 个参数
- 动态参数:
  - 可以在数据库联机时更改
  - 可以在以下级别更改:
    - 会话级别
    - 系统级别
  - 在会话持续期间有效, 或有效性取决于 SCOPE 设置
  - 使用 ALTER SESSION 和 ALTER SYSTEM 命令进行更改
  - 总共约 234 个参数



版权所有 © 2010, Oracle。保留所有权利。

### 更改初始化参数值

初始化参数有两种类型。

**静态参数:** 影响实例或整个数据库, 只能通过更改 init.ora 或 SPFILE 的内容来修改。  
静态参数要求关闭数据库后再重新启动数据库才能生效。无法对当前实例更改静态参数。

**动态参数:** 可以在数据库联机时更改。有两种类型的动态参数:

- 会话级别参数仅影响一个用户会话。这类参数的示例有国家语言支持 (NLS) 参数, 这些参数可用于指定排序、日期参数等的国家语言设置。这些参数可以在某个给定会话中使用, 并且会在该会话结束时失效。
- 系统级别参数影响整个数据库和所有会话。这类参数的示例有修改 SGA\_TARGET 值和设置归档日志目标位置。这些参数在指定的 SCOPE 内保持有效。要使这些参数设置永久有效, 必须通过指定 SCOPE=BOTH 选项或手动编辑 PFILE, 将这些参数添加到 SPFILE。

可以使用 ALTER SESSION 和 ALTER SYSTEM 命令更改动态参数。

## 更改初始化参数值（续）

使用 ALTER SYSTEM 语句的 SET 子句可以设置或更改初始化参数值。可选的 SCOPE 子句用于指定更改的作用域，如下所示：

- **SCOPE=SPFILE:** 此更改仅在服务器参数文件中应用。不会对当前实例进行更改。对于动态和静态参数，更改在下一次启动后生效，并且永久保持有效。对于静态参数来说，只允许指定该 SCOPE 值。
- **SCOPE=MEMORY:** 此更改仅应用到内存中。会对当前实例进行更改，且更改立即生效。对于动态参数，更改立即生效，但不会永久保持，因为服务器参数文件不会进行更新。对于静态参数，不允许指定该值。
- **SCOPE=BOTH:** 此更改会应用到服务器参数文件和内存中。会对当前实例进行更改，且更改立即生效。对于动态参数，更改永久保持有效，因为服务器参数文件会进行更新。对于静态参数，不允许指定该值。

如果实例不是使用服务器参数文件启动的，则指定 SCOPE=SPFILE 或 SCOPE=BOTH 是错误的。如果实例是使用服务器参数文件启动的，则默认值为 SCOPE=BOTH；如果实例是使用文本初始化参数文件启动的，则默认值为 MEMORY。

对于某些动态参数，还可以指定 DEFERRED 关键字。如果指定了该关键字，则更改仅对于以后的会话有效。这仅对以下参数有效：

- backup\_tape\_io\_slaves
- recyclebin
- audit\_file\_dest
- object\_cache\_optimal\_size
- object\_cache\_max\_size\_percent
- sort\_area\_size
- sort\_area\_retained\_size
- olap\_page\_pool\_size

如果将 SCOPE 指定为 SPFILE 或 BOTH，则使用可选的 COMMENT 子句可以将一个文本字符串与参数更新关联起来。此注释会写入服务器参数文件中。

## 更改参数值：示例

```
SQL> ALTER SESSION  
      SET NLS_DATE_FORMAT ='mon dd yyyy' ;  
  
Session altered.  
  
SQL> SELECT SYSDATE FROM dual;  
  
SYSDATE  
-----  
jun 18 2009
```

```
SQL> ALTER SYSTEM SET  
SEC_MAX_FAILED_LOGIN_ATTEMPTS=2 COMMENT='Reduce  
from 10 for tighter security.' SCOPE=SPFILE;  
  
System altered.
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 更改参数值：示例

幻灯片中的第一条语句是一个更改会话级别参数的示例。用户要将会话日期格式设置为 mon dd yyyy。因此，对此日期的所有查询将以该格式显示日期。还可以使用 PL/SQL 在应用程序中设置会话级别参数。

第二条语句更改断开连接之前的最大失败登录尝试次数。它包括一条注释，显式说明仅在服务器参数文件中进行这一更改。经过指定的失败尝试次数之后，服务器进程会自动断开连接。这不是一个动态参数，需要重新启动 Oracle DB 实例，更改才会生效。

## 小测验

Enterprise Manager Database Control 可用来对许多数据库进行并发管理。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小测验

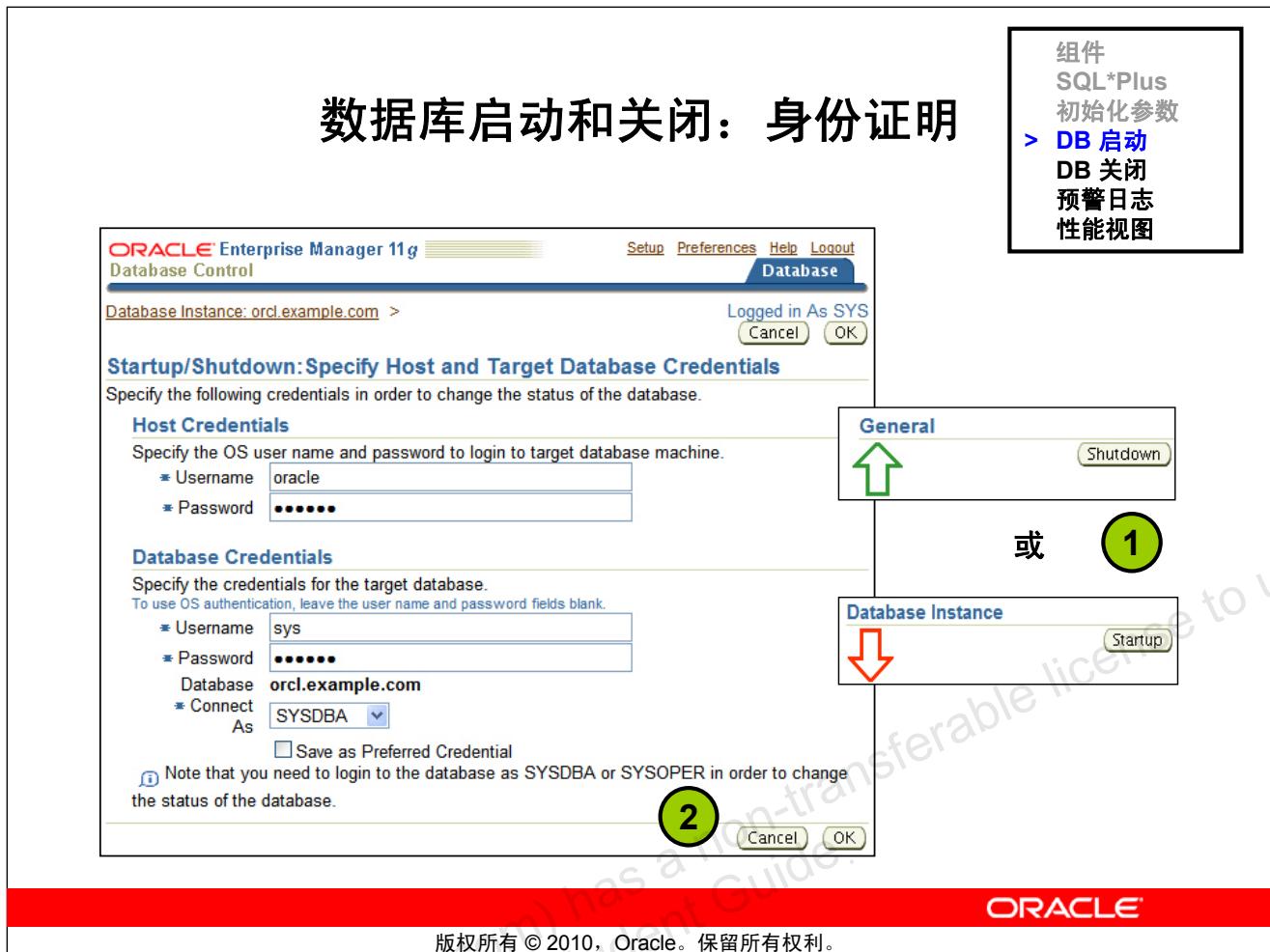
多数数据库参数为动态参数，不必关闭数据库实例即可更改。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 1

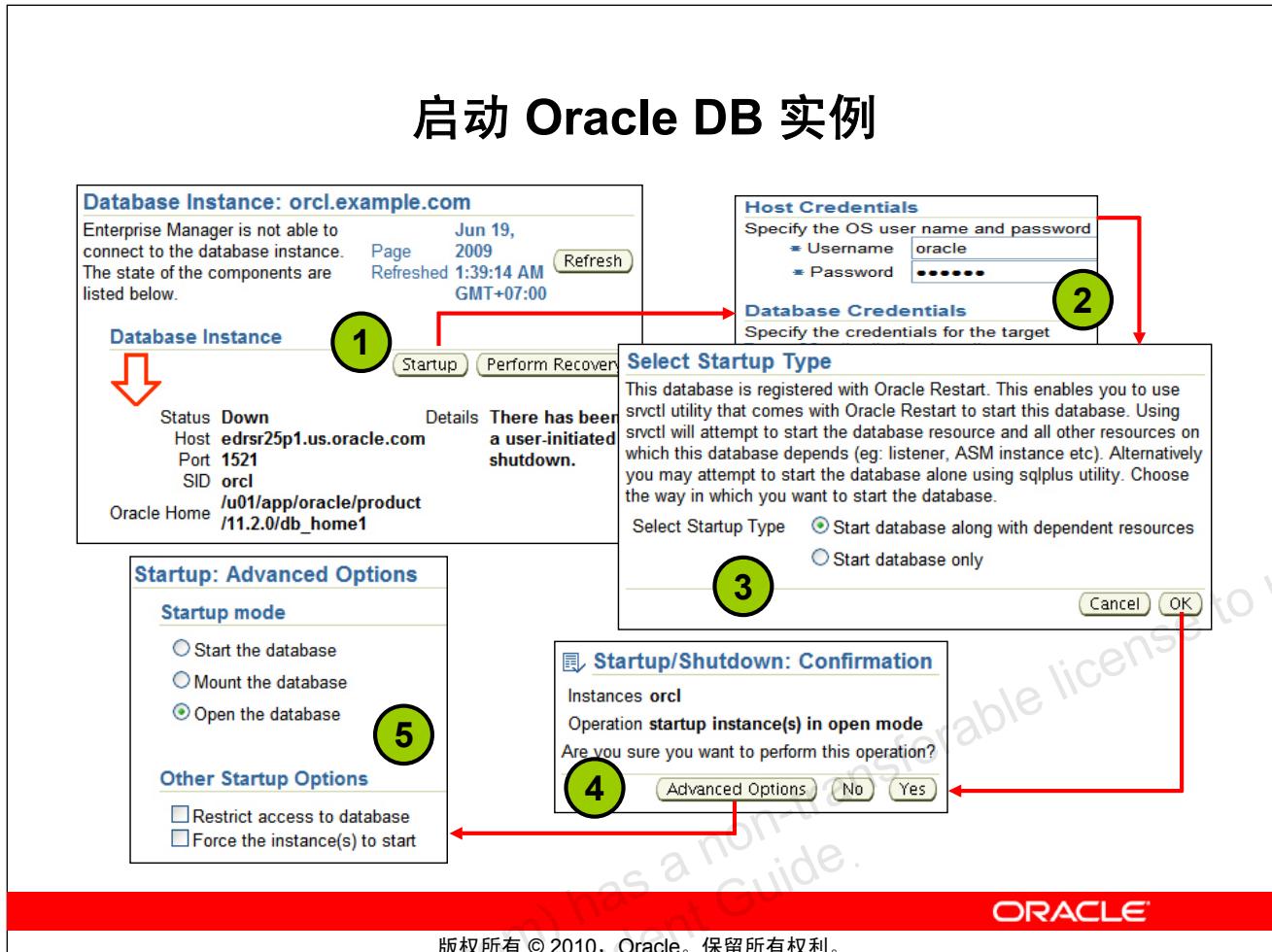


## 数据库启动和关闭：身份证明

单击“Startup（启动）”或“Shutdown（关闭）”时，系统会提示输入登录主机（数据库驻留的计算机）和登录数据库本身的的身份证明。必须输入具有 SYSDBA 权限的数据库帐户。然后输入身份证明。

输入身份证明信息之后，系统会提示您选择启动或关闭的方法。然后可以单击“Advanced Options（高级选项）”，根据需要更改任何启动选项或关闭模式。还可以单击“Show SQL（显示 SQL）”查看用于启动或关闭的 SQL 语句。

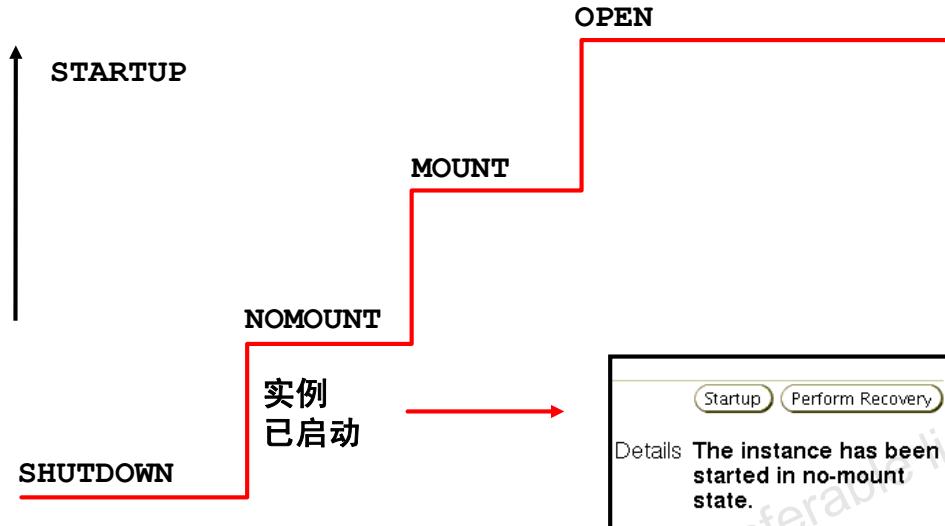
**注：**使用 Enterprise Manager 进行关闭时的默认选项是 IMMEDIATE。从 SQL\*Plus 发出 SHUTDOWN 命令时使用的默认选项是 NORMAL。



## 启动 Oracle DB 实例

如果转到 Enterprise Manager Database Control 页时数据库尚未启动，请单击“Startup（启动）”，然后输入主机身份证明，还可以选择启动模式。如果 Oracle DB 已经在 Oracle Restart 中注册，则会出现一个单独的对话框，提示您是选择使用 Server Control (SRVCTL) 实用程序还是 SQL\*Plus 来启动数据库实例。使用 Oracle Restart 时，建议使用 SRVCTL 实用程序，因为它可以启动那些可能会需要的相关资源。

## 启动 Oracle DB 实例: NOMOUNT



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 启动 Oracle DB 实例: NOMOUNT

启动数据库实例时，请选择实例启动时所处的状态。以下场景描述启动实例过程中的不同阶段。

在数据库创建期间、控制文件重新创建期间或执行特定备份和恢复方案期间，实例通常仅以 NOMOUNT 模式启动。

启动实例过程包括执行以下任务：

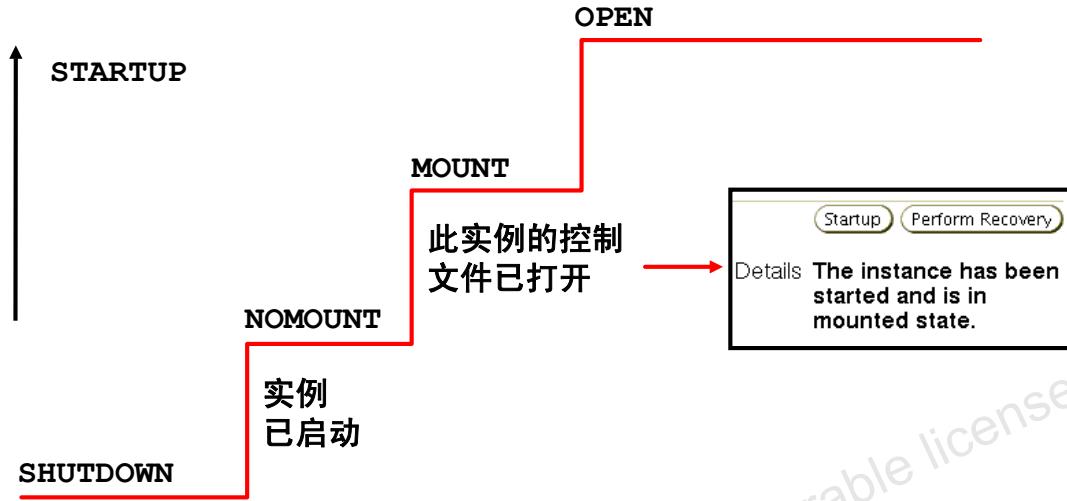
- 按以下顺序搜索 \$ORACLE\_HOME/dbs 中具有特定名称的文件：
  - 搜索 spfile<SID>.ora。
  - 如果未找到 spfile<SID>.ora，则搜索 spfile.ora。
  - 如果未找到 spfile.ora，则搜索 init<SID>.ora。

这是包含实例初始化参数的文件。使用 STARTUP 指定 PFILE 参数可覆盖默认行为。

- 分配 SGA
- 启动后台进程
- 打开 alert\_<SID>.log 文件和跟踪文件

注：SID 是用于标识实例名称的系统 ID（例如 ORCL）。

## 启动 Oracle DB 实例: MOUNT



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 启动 Oracle DB 实例: MOUNT

数据库装载过程包括执行以下任务:

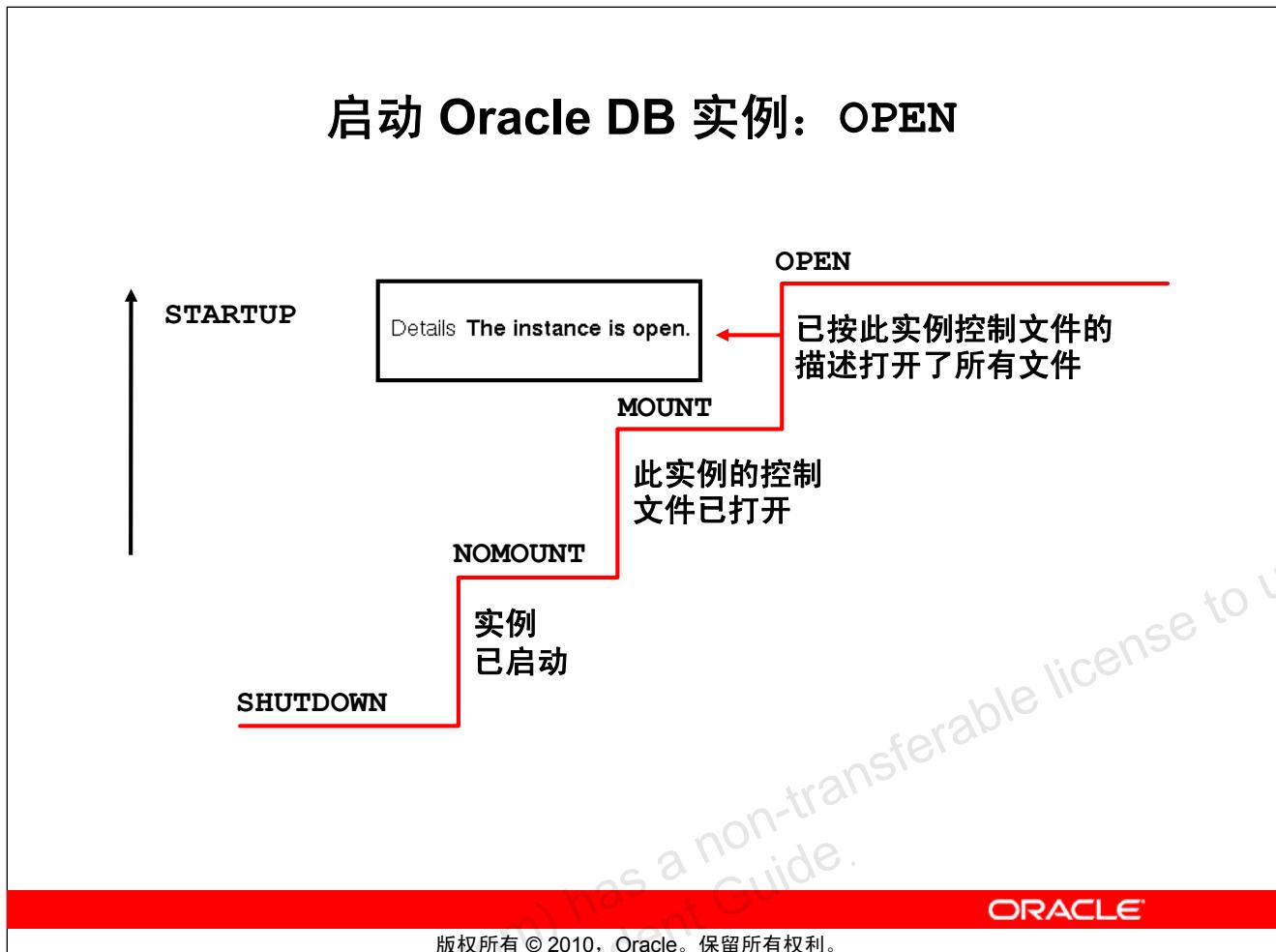
- 将数据库与之前启动的实例关联
- 定位并打开参数文件中指定的所有控制文件
- 通过读取控制文件来获取数据文件和联机重做日志文件的名称和状态（但是，此时不会执行检查来验证是否存在数据文件和联机重做日志文件。）

要执行特定的维护操作，请启动实例，然后装载数据库，但不打开该数据库。

例如，在执行以下任务期间必须装载数据库，但不得打开数据库:

- 重命名数据文件（打开数据库时可重命名脱机表空间的数据文件。）
- 启用和禁用联机重做日志文件归档选项
- 执行完整的数据库恢复

**注:** 即使发出了 OPEN 请求，数据库仍可能处于 MOUNT 模式下。这是因为可能需要以某种方式恢复数据库。如果在 MOUNT 状态下执行恢复，将打开重做日志进行读取，并且打开数据文件读取需要恢复的块，并在恢复期间根据需要写入块。



## 启动 Oracle DB 实例: OPEN

正常的数据库操作意味着实例已启动、数据库已装载且已打开。通过执行正常数据库操作，任何有效用户都可连接到数据库，而且可执行常规的数据访问操作。

打开数据库过程包括执行以下任务：

- 打开数据文件
- 打开联机重做日志文件

如果尝试打开数据库时任一数据文件或联机重做日志文件不存在，则 Oracle 服务器返回错误。

在最后这个阶段，Oracle 服务器会验证是否可以打开所有数据文件和联机重做日志文件，还会检查数据库的一致性。如有必要，系统监视器 (SMON) 后台进程将启动实例恢复。

可以在受限模式下启动数据库实例，使得只让有管理权限的用户使用该实例。要在受限模式下启动实例，请在“Advanced Startup Options (高级启动选项)”页上选择“Restrict access to database (限制对数据库进行访问)”选项。

## 启动选项：示例

- 使用 sqlplus 实用程序：

**SQL> startup**

**1**

**SQL> startup nomount**

**2**

**SQL> alter database mount;**

**3**

**SQL> alter database open;**

**4**

- 启用 Oracle Restart 时，可使用 srvctl 实用程序。

**\$ srvctl start database -d orcl -o mount**

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## 启动选项：示例

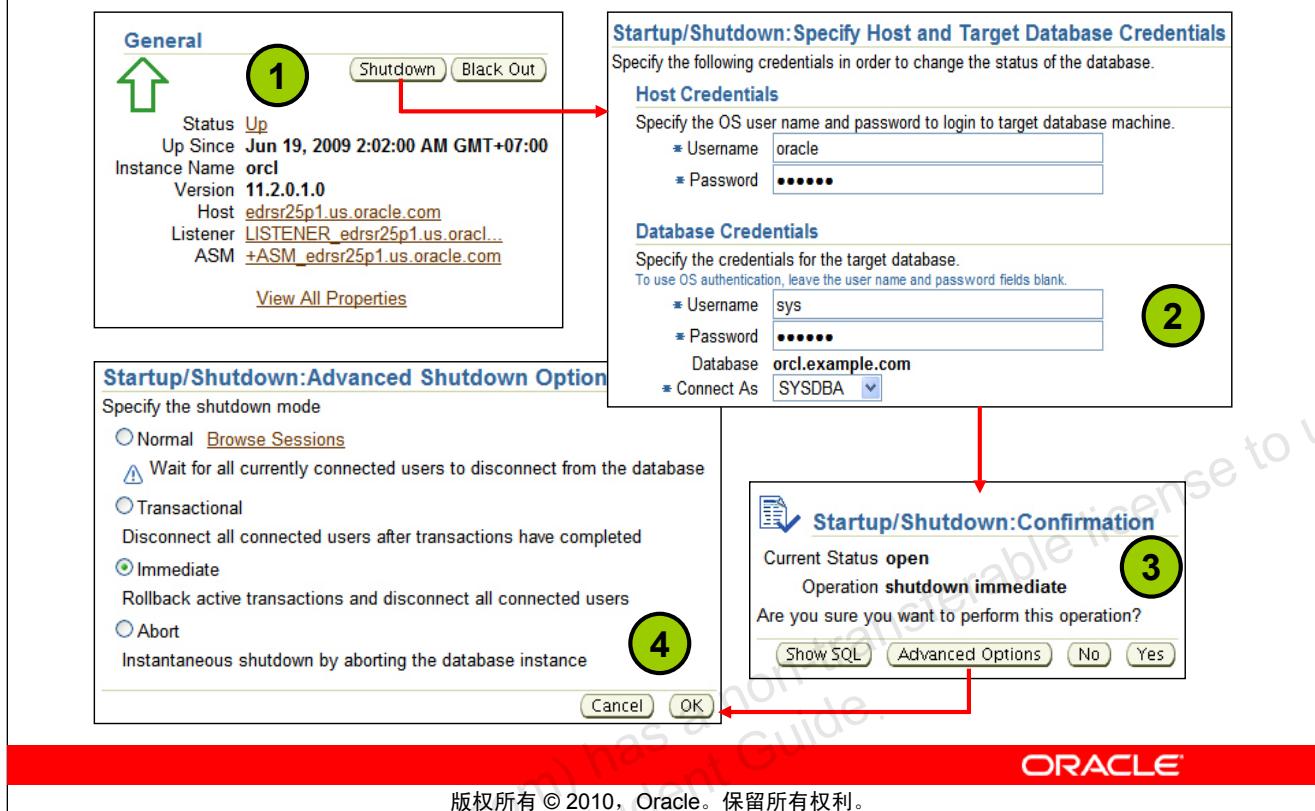
此幻灯片显示用于启动数据库的 SQL\*Plus 语法。

- 该命令启动实例，将数据库文件与该实例关联，然后装载并打开数据库。
- 该命令启动实例，但不装载数据库。
- 该命令装载 NOMOUNT 状态的数据库。
- 该命令打开 MOUNT 状态的数据库。

当数据库启用 Oracle Restart 时，可以使用 srvctl 实用程序启动数据库实例。srvctl 实用程序的优势在于，它还可以启动所有必需的相关资源，如 ASM 实例、ASM 磁盘组和监听程序。

**注：**srvctl 实用程序位于 Grid Infrastructure 软件的 \$ORACLE\_HOME/bin 目录和 Oracle DB 软件的 \$ORACLE\_HOME/bin 目录中。启动 Oracle DB 时，应使用 Oracle DB 软件中的 srvctl 实用程序。启动 ASM 实例或监听程序时，应使用 Grid Infrastructure 软件中的 srvctl 实用程序。

## 关闭 Oracle DB 实例



### 关闭 Oracle DB 实例

如果转到 Enterprise Manager Database Control 页时实例已启动，则可以单击“Shutdown（关闭）”按钮关闭该实例。系统将提示您验证或输入主机和数据库的身份证明。单击“OK（确定）”会出现“Startup/Shutdown: Confirmation（启动/关闭：确认）”对话框。之后，如果单击“Advanced Options（高级选项）”按钮，可以选择关闭模式：NORMAL、TRANSACTIONAL、IMMEDIATE 或 ABORT。

## 关闭模式

关闭模式	A	I	T	N
允许新连接	否	否	否	否
等待当前会话结束	否	否	否	是
等待当前事务处理结束	否	否	是	是
强制选择检查点并关闭文件	否	是	是	是

### 关闭模式:

- A = ABORT
- I = IMMEDIATE
- T = TRANSACTIONAL
- N = NORMAL

ORACLE

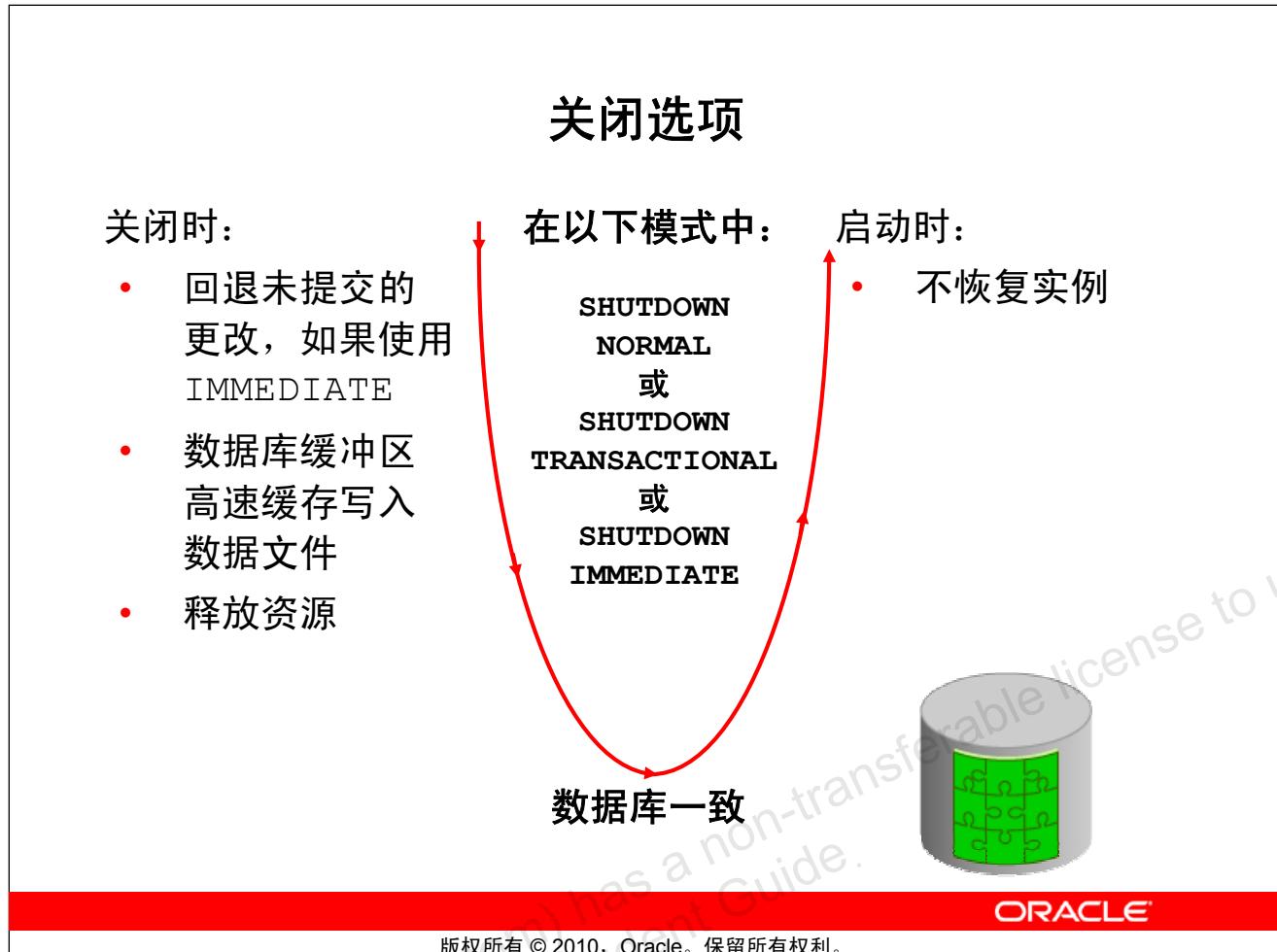
版权所有 © 2010, Oracle。保留所有权利。

### 关闭模式

关闭模式对当前活动的适用性按以下顺序逐渐增强:

- **ABORT:** 在关闭之前执行最少量的任务。由于此模式需要在启动之前进行恢复，因此只在需要时才使用此模式。当启动实例时出现了问题，或者因紧急情况（如，通知在数秒内断电）而需要立即关闭时，如果其它关闭方式都不起作用，通常选择使用此模式。
- **IMMEDIATE:** 这是最常用选项。选择此模式会回退未提交的事务处理。
- **TRANSACTIONAL:** 允许完成现有事务处理，但不启动新的事务处理。
- **NORMAL:** 等待会话断开。

如果考虑执行关闭所花费的时间，则会发现 ABORT 的关闭速度最快，而 NORMAL 的关闭速度最慢。NORMAL 和 TRANSACTIONAL 花费的时间较长，具体取决于会话和事务处理的数目。



## 关闭选项

### SHUTDOWN NORMAL

如果不指定模式, 则默认关闭模式为 NORMAL。正常关闭数据库时会发生以下情况:

- 不可以建立新连接。
- Oracle 服务器在完成关闭之前先等待所有用户断开连接。
- 数据库和重做缓冲区被写入磁盘。
- 后台进程终止, 从内存中删除 SGA。
- Oracle 服务器在关闭实例之前关闭并卸装数据库。
- 下一次启动不需要进行实例恢复。

### SHUTDOWN TRANSACTIONAL

采用 TRANSACTIONAL 关闭方式可防止客户机丢失数据, 其中包括客户机当前活动的结果。执行事务处理数据库关闭时会发生以下情况:

- 任何客户机都不能在这个特定实例上启动新事务处理。
- 会在客户机结束正在进行的事务处理后断开客户机。
- 完成所有事务处理后立即执行关闭。
- 下一次启动不需要进行实例恢复。

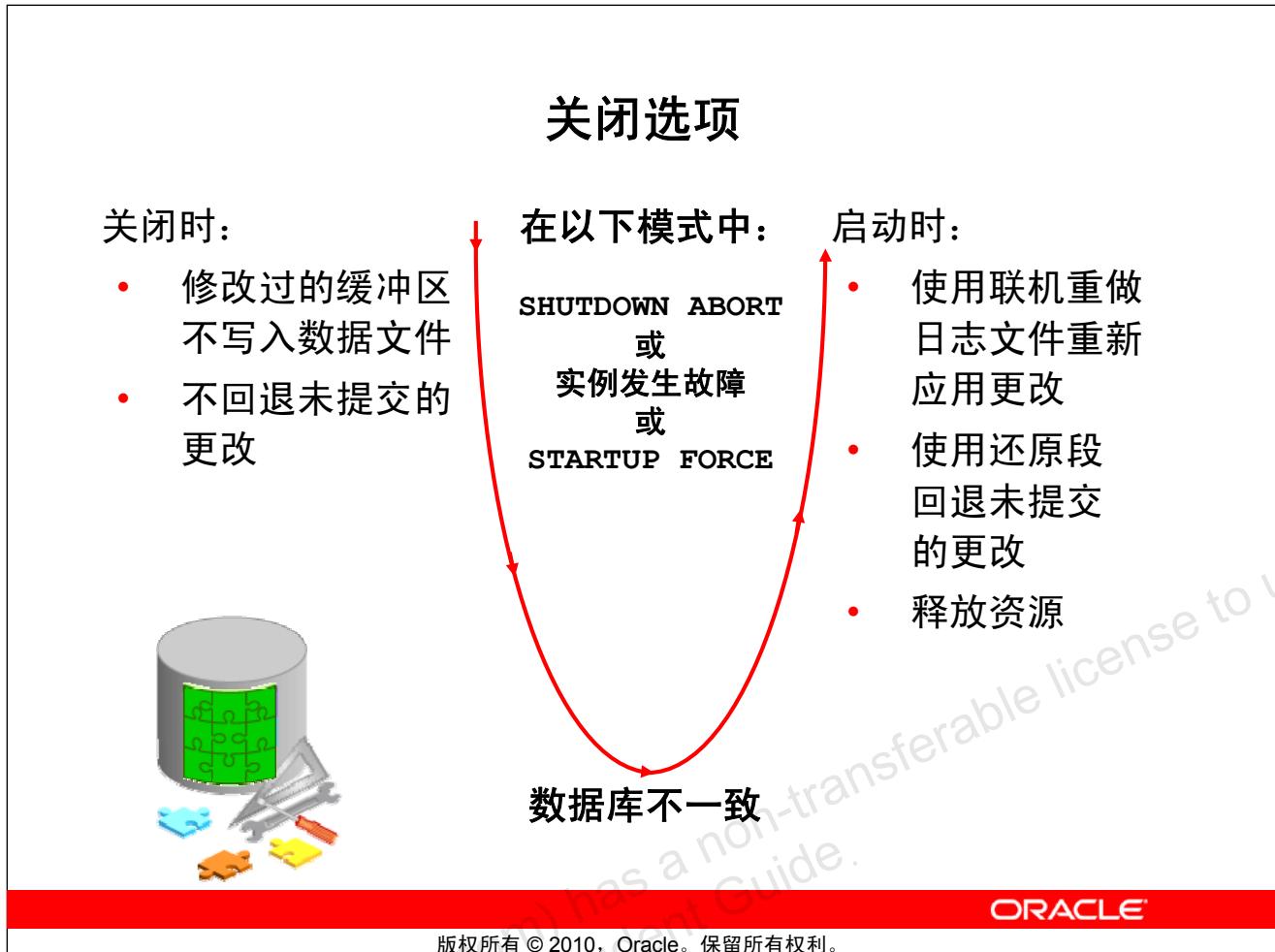
## 关闭选项（续）

### **SHUTDOWN IMMEDIATE**

采用 IMMEDIATE 关闭模式会出现以下情况：

- Oracle DB 正在处理的当前 SQL 语句不会完成执行。
- Oracle 服务器不会等待当前连接到数据库的用户断开连接。
- Oracle 服务器会回退活动的事务处理，并会断开所有连接用户。
- Oracle 服务器在关闭实例之前关闭并卸装数据库。
- 下一次启动不需要进行实例恢复。

注：IMMEDIATE 是使用 Enterprise Manager 时的默认关闭模式。



### 关闭选项（续）

#### **SHUTDOWN ABORT**

如果 NORMAL、TRANSACTIONAL 和 IMMEDIATE 关闭模式都不起作用，则可以选择中止当前的数据库实例。中止实例时会发生以下情况：

- Oracle DB 正在处理的当前 SQL 语句会立即终止。
- Oracle 服务器不等待当前连接到数据库的用户断开连接。
- 数据库和重做缓冲区未写入磁盘。
- 不回退未提交的事务处理。
- 实例终止，但不关闭文件。
- 数据库未关闭或未卸装。
- 下一次启动时需要进行实例恢复，实例恢复是自动进行的。

注：不建议备份处于不一致状态的数据库。

## 关闭选项：示例

- 使用 SQL\*Plus

**SQL> shutdown**

1

**SQL> shutdown transactional**

2

**SQL> shutdown immediate**

3

**SQL> shutdown abort**

4

- 启用 Oracle Restart 时，可使用 SRVCTL 实用程序

**\$ srvctl stop database -d orcl -o abort**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 关闭选项：示例

幻灯片显示的是使用 SQL\*Plus 和 SRVCTL 实用程序关闭数据库的示例。

- 该命令启动正常关闭。在所有用户注销之前，不会关闭数据库。
- 该命令启动事务处理关闭。在完成所有现有事务处理之前，数据库不会关闭。
- 该命令启动立即关闭。将回退未提交的事务处理。
- 该命令启动中止关闭。

当数据库启用 Oracle Restart 时，可以使用 SRVCTL 实用程序关闭数据库实例。

**注：**SRVCTL 实用程序位于 Grid Infrastructure 软件的 \$ORACLE\_HOME/bin 目录和 Oracle DB 软件的 \$ORACLE\_HOME/bin 目录中。启动 Oracle DB 时，应使用 Oracle DB 软件中的 SRVCTL 实用程序。启动 ASM 实例或监听程序时，应使用 Grid Infrastructure 软件中的 SRVCTL 实用程序。

## 查看预警日志

“Database (数据库) ”主页 > “Related Links (相关链接) ”区域 > “Alert Log Content (预警日志内容) ”

组件  
SQL\*Plus  
初始化参数  
DB 启动  
DB 关闭  
> 预警日志  
性能视图

View Entries Last 50 Go Search					
Timestamp	Type ▾	Level	Incident ID/Group	Message ID	Message Text
Jun 19, 2009 10:00:16 PM GMT+07:00	NOTIFICATION	16	sqltune	kesaiTuneSqlDrv:5067.3456118459	End automatic SQL Tuning Advisor run for special tuning task "SYS_AUTO_SQL_TUNING_TASK"
Jun 19, 2009 10:00:03 PM GMT+07:00	NOTIFICATION	16	sqltune	kesaiTuneSqlDrv:4555.2579917519	Begin automatic SQL Tuning Advisor run for special tuning task "SYS_AUTO_SQL_TUNING_TASK"
Jun 19, 2009 10:00:00 PM GMT+07:00	NOTIFICATION	16	process start	ksbrdp:3833:3697353022	VKRM started with pid=24, OS id=7929
Jun 19, 2009 10:00:00 PM GMT+07:00	NOTIFICATION	16	process start	ksbs1p_real:2253:2371767696	Starting background process VKRM
Jun 19, 2009 2:07:22 AM GMT+07:00	NOTIFICATION	16	process start	ksbrdp:3833:3697353022	SMCO started with pid=23, OS id=30582
Jun 19, 2009 2:07:22 AM GMT+07:00	NOTIFICATION	16	process start	ksbs1p_real:2253:2371767696	Starting background process SMCO
Jun 19, 2009 2:02:26 AM GMT+07:00	NOTIFICATION	16	process start	ksbrdp:3833:3697353022	CJQ0 started with pid=33, OS id=29846

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 查看预警日志

每个数据库都有一个 `alert_<sid>.log` 文件。此文件位于数据库所在的服务器中，如果设置了 `$ORACLE_BASE`，则此文件默认存储在 `$ORACLE_BASE/diag/rdbms/<db_name>/<SID>/trace` 中。

数据库预警文件是按时间顺序列出消息的日志文件，例如：

- 启动时使用的任何非默认初始化参数
- 已发生的所有内部错误 (ORA-600)、块损坏错误 (ORA-1578) 和死锁错误 (ORA-60)
- 管理操作，如 SQL 语句 CREATE、ALTER、DROP DATABASE 和 TABLESPACE，以及 Enterprise Manager 或 SQL\*Plus 语句 STARTUP、SHUTDOWN、ARCHIVE LOG 和 RECOVER
- 与共享服务器和分派程序进程的功能相关的多个消息和错误
- 自动刷新实体化视图时发生的错误

Oracle DB 使用预警日志来保留这些事件的记录，以此作为在操作员控制台上显示这些信息的替代方法。（许多系统会同时在控制台中显示这些信息。）如果某个管理操作成功完成，系统会将“completed (已完成)”消息和一个时间戳写入预警日志中。

## 查看预警日志（续）

Enterprise Manager 可监视预警日志文件，并向您通知严重的错误。您还可以查看日志，以检查不严重的错误和参考性消息。由于预警文件会增长到无法管理的大小，因此可以定期备份该文件，并删除当前的预警文件。当数据库尝试再次写入预警文件时，会重新创建一个新预警文件。

**注：**\$ORACLE\_BASE/diag/rdbms/<db\_name>/<SID>/alert 目录中有一个 XML 版本的预警日志。

**要通过 SQL\*Plus 确定预警日志的位置，请执行以下操作：**

- 使用 SQL\*Plus（或其它查询工具，如 SQL Developer）连接到数据库。
- 查询 V\$DIAG\_INFO 视图。

**要查看不带 XML 标记的纯文本预警日志，请执行以下操作：**

- 在 V\$DIAG\_INFO 查询结果中，记下对应于 Diag Trace 条目的路径。将目录更改至该路径。
- 使用文本编辑器打开 alert\_SID.log 文件。

**要查看 XML 格式的预警日志，请执行以下操作：**

- 在 V\$DIAG\_INFO 查询结果中，记下对应于 Diag Alert 条目的路径。将目录更改至该路径。
- 使用文本编辑器打开 log.xml 文件。

## 使用跟踪文件

- 每个服务器和后台进程都会向关联的跟踪文件写入信息。
- 错误信息写入相应的跟踪文件中。
- 自动诊断资料档案库 (ADR)。
  - 包含整个系统的跟踪和事件记录信息的中央资料档案库
  - 存储数据库诊断数据，例如：
    - 跟踪
    - 预警日志
    - 健康状况监视器报告



版权所有 © 2010, Oracle。保留所有权利。

### 使用跟踪文件

每个服务器和后台进程都会向关联的跟踪文件写入信息。进程检测到内部错误时，会将有关错误的信息转储到进程的跟踪文件。如果发生了内部错误并且在跟踪文件中写入了信息，则管理员应与 Oracle 支持服务部门联系。

与后台进程关联的跟踪文件的所有文件名均包含生成该跟踪文件的进程的名称。唯一的例外是由作业队列进程 (Jnnn) 生成的跟踪文件。

跟踪文件中的附加信息可以提供用于优化应用程序或实例的指导信息。后台进程总会在适当的时候将该信息写入跟踪文件中。

从 Oracle Database 11g 起，引入了一个高级错误诊断基础结构，用于避免、检测、诊断和解决问题。特别是此功能所针对的问题包含严重错误，例如由数据库代码 Bug、元数据损坏和客户数据损坏导致的严重错误。

## 使用跟踪文件（续）

发生严重错误时，会为其分配一个意外事件编号；系统将立即捕获该错误的诊断数据（如跟踪文件），并使用该编号标记此数据。该数据随后存储在自动诊断资料档案库（ADR，数据库之外的基于文件的资料档案库）中，以后可以通过意外事件编号在其中检索和分析该数据。

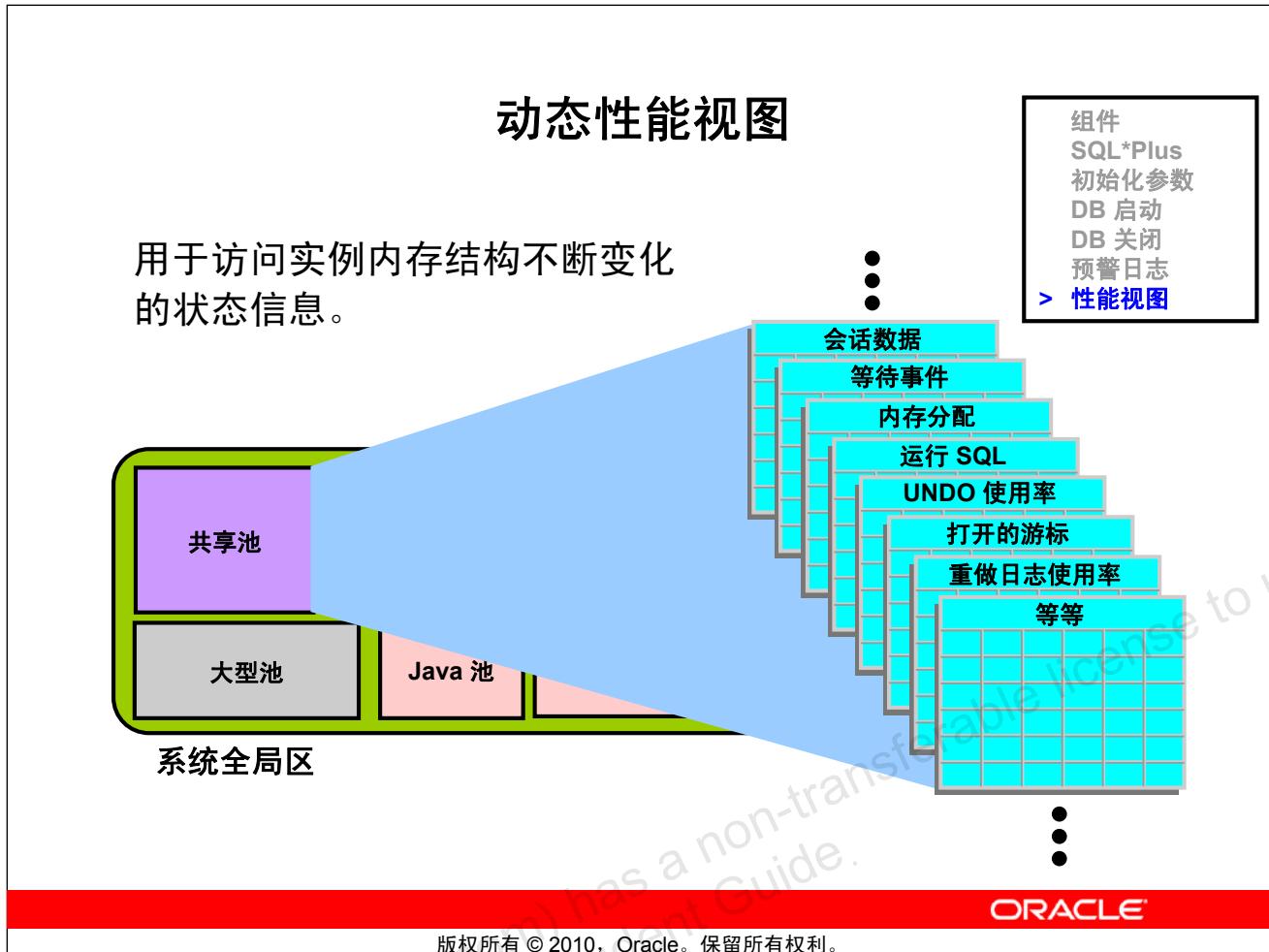
ADR 是包含整个系统的跟踪和事件记录信息的中央资料档案库，用于提供数据库诊断数据（如跟踪、预警日志、健康状况监视器报告等）。

ADR 根目录又称为 ADR 基目录，其位置由 DIAGNOSTIC\_DEST 初始化参数设置。如果省略该参数或将其留空，则数据库在启动时按如下所示设置 DIAGNOSTIC\_DEST：

- 如果设置了 ORACLE\_BASE 环境变量，则会将 DIAGNOSTIC\_DEST 设置为由 ORACLE\_BASE 指定的目录。
- 如果未设置 ORACLE\_BASE 环境变量，则将 DIAGNOSTIC\_DEST 设置为 \$ORACLE\_HOME/log。

ADR 主目录的位置由以下路径给定，该路径以 ADR 基目录开头：

```
./diag/product_type/db_id/instance_id
```



## 动态性能视图

在 Oracle DB 中还维护着一个关于数据库实例操作和性能的动态数据集。这些动态性能视图以基于数据库服务器内部的内存结构构建的虚拟表为基础。也就是说，这些视图不是驻留在数据库中的传统意义上的表。这就是为什么部分视图在数据库装载或打开之前就能够使用的原因。

动态性能视图包含下列信息：

- 会话
- 文件状态
- 作业和任务的进度
- 锁
- 备份状态
- 内存使用率和内存分配
- 系统参数和会话参数
- SQL 执行
- 统计数据和度量

注：DICT 和 DICT\_COLUMNS 视图也包含这些动态性能视图的名称。动态性能视图以前缀“v\$”开头，其数量超过 590。

## 动态性能视图：用法示例

1

```
SQL> SELECT sql_text, executions FROM v$sql  
WHERE cpu_time > 200000;
```

2

```
SQL> SELECT * FROM v$session WHERE machine =  
'EDRSR9P1' and logon_time > SYSDATE - 1;
```

3

```
SQL> SELECT sid, ctime FROM v$lock  
WHERE block > 0;
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 动态性能视图：用法示例

经常使用这些视图的是 Enterprise Manager，但其他用户也可以根据需要查询这些视图。

幻灯片中所示的三个示例回答了下列问题：

1. 哪些 SQL 语句（及其相关的执行编号）所占用的 CPU 时间大于 200,000 微秒？
2. 在前一天内，哪些当前会话从 EDRSR9P1 计算机登录？
3. 当前持有阻止其他用户的锁的那些会话的会话 ID 是什么？这些锁已持有多长时间？

## 动态性能视图：注意事项

- 这些视图由 SYS 用户所有。
- 不同场合可使用的视图也不同：
  - 实例已启动。
  - 数据库已装载。
  - 数据库已打开。
- 可以通过查询 V\$FIXED\_TABLE 查看所有视图名称。
- 这些视图通常称为“v\$ 视图”。
- 因为其数据是动态的，所以在这些视图上不能保证读取一致性。

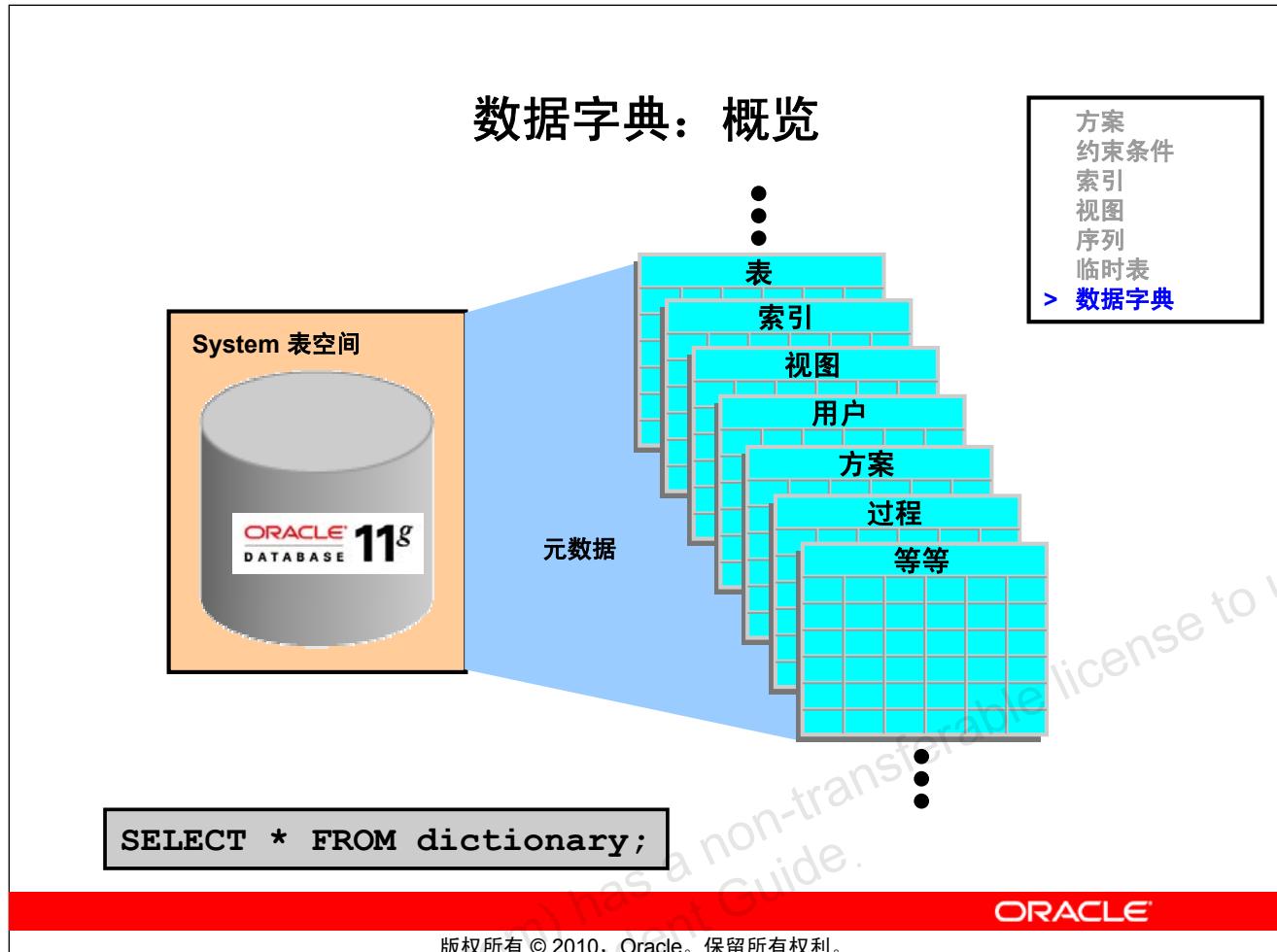


版权所有 © 2010, Oracle。保留所有权利。

### 动态性能视图：注意事项

某些动态视图包含的数据并不适用于实例或数据库的所有状态。例如，如果实例刚启动，但未装载任何数据库，则可以通过查询 V\$BGPROMESS 来查看正在运行的后台进程的列表。但无法通过查询 V\$DATAFILE 查看数据库数据文件的状态，因为要通过装载数据库来读取控制文件，此后才能找出数据库关联的数据文件。

一些 v\$ 视图包含的信息类似于相应 DBA\_ 视图中的信息。例如，V\$DATAFILE 类似于 DBA\_DATA\_FILES。另请注意，v\$ 视图名称通常是单数形式，而 DBA\_ 视图名称是复数形式。



## 数据字典：概览

Oracle 数据字典是数据库的元数据，其中包含数据库中所有对象的名称和属性。创建或修改任何对象都会引起数据字典的更新，以反映这些更改。这些信息存储在由 Oracle DB 维护的基表中，不过，可使用预定义视图访问这些表（而不是直接读取表）。

数据字典：

- 由 Oracle DB 服务器使用，可用于查找关于用户、对象、约束条件和存储的信息
- 对象结构或定义修改时，Oracle DB 将做出相应维护
- 可供任意用户使用以查询关于数据库的信息
- 由 SYS 用户拥有
- 不得使用 SQL 直接进行修改

**注：**DICTIONARY 数据字典视图（或其 DICT 同义词）包含数据字典表和视图的名称和说明。使用 DICT\_COLUMNS 视图可查看视图列及其定义。有关各种视图的完整定义，请参阅《Oracle Database Reference》。共有 1000 多个引用数百个基表的视图。

## 数据字典视图

	谁可以查询	内容	下列项的子集	附注
DBA_	DBA	一切对象	N/A	可能有其它仅限 DBA 使用的列
ALL_	每个用户	用户有权查看的一切对象	DBA_ 视图	包括用户自己的对象以及该用户已获得查看权限的其它对象
USER_	每个用户	用户拥有的 一切对象	ALL_ 视图	通常与 ALL_ 相同，只是缺少 OWNER 列（某些视图使用缩写名称作为 PUBLIC 的同义词。）

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 数据字典视图

视图前缀指示给定用户可以看到的数据（以及数据量）。

只有拥有 DBA 权限的用户才可访问包含所有对象的全局视图（带 DBA\_ 前缀）。

下一个权限级别是 ALL\_ 前缀级别，它表示查询用户有权查看的所有对象，无论用户是否拥有这些对象。例如，如果 USER\_A 被授权访问属于 USER\_B 的表，那么 USER\_A 会看到该表列在与表名有关的所有 ALL\_ 视图中。

前缀 USER\_ 代表最小的可见范围。该类型的视图只显示属于查询用户的那些对象（即存在于用户自己方案中的对象）。

一般情况下，每个视图集都是拥有更高权限的视图集的子集，无论按行还是按列。并非给定视图集中的所有视图都在其它视图集中有对应的视图。这取决于视图中信息的性质。例如，有一个 DBA\_LOCK 视图，但是并不存在 ALL\_LOCK 视图。这是因为只有 DBA 才关注有关锁的数据。为了满足需要，应确保选择适当的视图集。即便您有权访问 DBA 视图，也可能只想查询 USER 版本的视图，因为结果会显示您所拥有的对象的相关信息，而您可能不希望结果集中包含其他对象。

## 数据字典视图（续）

DBA\_ 视图只能由拥有 SYSDBA 或 SELECT ANY DICTIONARY 权限的用户查询。

并非所有字典视图都以前缀 DBA\_、ALL\_ 和 USER\_ 开头。以下视图或视图的同义词就是例外：

- AUDIT\_ACTIONS
- CAT
- CHANGE\_PROPAGATIONS
- CHANGE\_PROPAGATION\_SETS
- CHANGE\_SETS
- CHANGE\_SOURCES
- CHANGE\_TABLES
- CLIENT\_RESULT\_CACHE\_STATS\$
- CLU
- COLS
- COLUMN\_PRIVILEGES
- DATABASE\_COMPATIBLE\_LEVEL
- DBMS\_ALERT\_INFO
- DBMS\_LOCK\_ALLOCATED
- DICT
- DICTIONARY
- DICT\_COLUMNS
- DUAL
- GLOBAL\_NAME
- IND
- INDEX\_HISTOGRAM
- INDEX\_STATS
- LOGSTDBY\_UNSUPPORTED\_TABLES
- NLS\_DATABASE\_PARAMETERS
- NLS\_INSTANCE\_PARAMETERS
- NLS\_SESSION\_PARAMETERS
- OBJ
- RECYCLEBIN
- RESOURCE\_COST
- ROLE\_ROLE\_PRIVS
- ROLE\_SYS\_PRIVS
- ROLE\_TAB\_PRIVS
- SEQ
- SESSION\_PRIVS
- SESSION\_ROLES
- SM\$VERSION
- SYN
- TABLE\_PRIVILEGES
- TABS

## 数据字典：用法示例

1

```
SELECT table_name, tablespace_name
FROM user_tables;
```

2

```
SELECT sequence_name, min_value, max_value,
increment_by
FROM all_sequences
WHERE sequence_owner IN ('MDSYS', 'XDB');
```

3

```
SELECT USERNAME, ACCOUNT_STATUS
FROM dba_users
WHERE ACCOUNT_STATUS = 'OPEN';
```

4

```
DESCRIBE dba_indexes
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据字典：用法示例

幻灯片中的示例查询回答了下列问题：

1. 方案中已创建的表的名称（以及表所在表空间的名称）是什么？
2. 在您有权访问的数据库中，关于序列的重要信息是什么？
3. 此数据库中当前可以登录的用户是哪些？
4. DBA\_INDEXES 视图的列是什么？这显示了您可以查看数据库中所有索引的哪些信息。

下面列出了此命令的部分输出：

Name	Null?	Type
OWNER	NOT NULL	VARCHAR2(30)
INDEX_NAME	NOT NULL	VARCHAR2(30)
INDEX_TYPE		VARCHAR2(27)
TABLE_OWNER	NOT NULL	VARCHAR2(30)
TABLE_NAME	NOT NULL	VARCHAR2(30)

## 小测验

使用 Oracle Restart 时，必须使用 Server Control 实用程序 (srvctl) 来启动和停止数据库实例，而不能使用 SQL\*Plus。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小测验

下列哪个数据字典视图可以用来查找数据库中所有表的名称？

1. USER\_TABLES
2. ALL\_TABLES
3. DBA\_TABLES
4. ANY\_TABLES

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 3

## 小结

在本课中，您应该已经学会：

- 启动和停止 Oracle DB 和组件
- 使用 Oracle Enterprise Manager
- 使用 SQL\*Plus 访问数据库
- 修改数据库初始化参数
- 描述数据库启动阶段
- 描述数据库关闭选项
- 查看预警日志
- 访问动态性能视图

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 4 概览：管理 Oracle 实例

本练习包含以下主题：

- 在 Enterprise Manager 中导航
- 查看和修改初始化参数
- 停止和启动数据库实例
- 查看预警日志
- 使用 SQL\*Plus 连接到数据库

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 管理 ASM 实例

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

华周 (286991486@qq.com) has a non-transferable license to use  
this Student Guide.

## 课程目标

学完本课后，应能完成以下工作：

- 描述使用 ASM 的好处
- 管理 ASM 实例
- 创建和删除 ASM 磁盘组
- 扩展 ASM 磁盘组
- 通过使用各种实用程序检索 ASM 元数据

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 课程目标

本课更加详细深入地讲述了 ASM 实例以及如何使用各种实用程序来管理该实例。

## ASM 对于管理员的好处

### 使用 ASM 可以免除:

- I/O 性能优化
- 数据文件移动和重新组织
- 文件名管理
- 逻辑卷管理
- 文件系统管理
- 集群文件系统管理
- 裸设备管理

### 使用 ASM 可以显著减少:

- 逻辑单元号 (LUN) 管理
  - 逻辑单元数量较少，大小较大
- 数据库管理员对系统管理员的依赖性
- 手动执行维护任务时可能发生的错误

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## ASM 对于管理员的好处

使用 ASM 可以免除非 ASM 存储环境中许多必不可少的任务。其中包括:

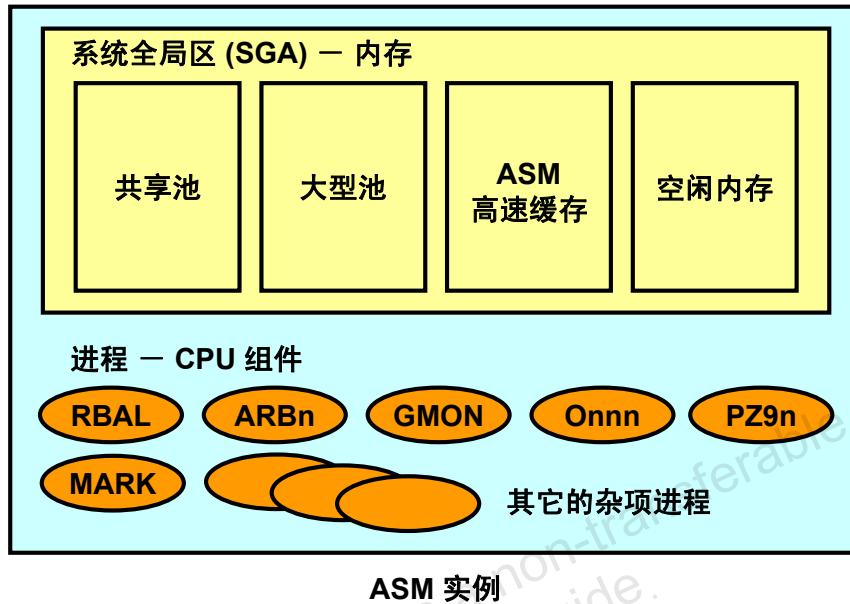
- **I/O 性能优化:** ASM 采用条带化和镜像所有内容的策略，且执行自动重新平衡操作，这意味着不再需要旨在平衡磁盘使用以及消除磁盘热点的 I/O 性能优化。
- **数据文件移动和重新组织:** 不再需要更改数据文件的位置来满足性能要求和空间约束条件。
- **文件名管理:** 不再需要定义和强制执行文件命名策略。
- **逻辑卷、文件系统、集群文件系统和裸设备管理:** 不再需要这些存储元素。

使用 ASM 可以减少下列重要方面的工作，从而提供更多好处:

- 逻辑单元号 (LUN) 管理工作减少，因为 ASM 通常需要的逻辑单元较少且大小较大。
- 数据库管理员与系统管理员之间通常存在的依赖性将大大减少。例如，添加新数据文件或将磁盘资源从一个磁盘组移至另一磁盘组时不再需要系统管理员干预。
- 手动执行维护任务时可能发生的错误将大大减少。例如，使用常规文件系统时，新建数据文件时可能不慎违反了文件命名惯例，导致数据库的其余部分不支持该文件。

## ASM 实例

ASM 实例是 ASM 的进程和内存组件的组合。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### ASM 实例

每次启动 ASM 或数据库时，都会分配名为系统全局区 (SGA) 的共享内存区域并启动 Oracle ASM 或数据库后台进程。后台进程和 SGA 的组合称为 Oracle ASM 实例或 Oracle DB 实例。该实例表示正在运行的 ASM 环境的 CPU 和 RAM 组件。

ASM 实例中的 SGA 与数据库实例中的 SGA 在内存分配和使用方面是不同的。ASM 实例中的 SGA 分为四个主要区域，如下所示：

- **共享池:** 用于元数据信息
- **大型池:** 用于并行操作
- **ASM 高速缓存:** 用于在重新平衡操作期间读取和写入块
- **空闲内存:** 可用的未分配内存

ASM 的建议最低内存量为 256 MB。ASM 实例默认启用自动内存管理，该功能将动态优化各个 SGA 内存组件的大小。ASM 实例所需的内存量将取决于 ASM 管理的磁盘空间量。

ASM 实例的第二部分是后台进程。ASM 实例可以具有许多后台进程；并不是所有进程始终都会出现。

## ASM 组件：ASM 实例（续）

下一幻灯片将讲述特定于 ASM 功能的后台进程。后台进程分为必需和可选两种。其中一些进程如下所示：

- **ARC $n$ :** 归档进程
- **CKPT:** 检查点进程
- **DBW $n$ :** 数据库写进程
- **DIAG:** 诊断进程
- **Jnnn:** 作业队列进程
- **LGWR:** 日志写进程
- **PMON:** 进程监视器进程
- **PSP0:** 进程衍生进程
- **QMN $n$ :** 队列监视器进程
- **RECO:** 恢复器进程
- **SMON:** 系统监视器进程
- **VKTM:** 虚拟计时器进程
- **MMAN:** 内存管理器进程

以上进程列表不是完整列表。对于 ASM 实例，这些进程并不总是执行它们在数据库实例中执行的任务。例如，数据库实例中的 LGWR 进程负责将更改向量从 SGA 的日志缓冲区部分复制到磁盘上的联机重做日志。ASM 实例的 SGA 中不包含日志缓冲区，该实例也不使用联机重做日志。ASM 实例中的 LGWR 进程将事件记录信息复制到 ASM 磁盘组。

如果 ASM 是以集群方式建立的，则将在 ASM 实例中运行与集群管理相关的附加进程。其中一些进程如下所示：

- **LMON:** 全局入队服务监视器进程
- **LMD $n$ :** 全局入队服务守护程序
- **LMS $n$ :** 全局高速缓存服务进程
- **LCK $n$ :** 锁定进程

## ASM 组件：ASM 实例 – 主要进程

ASM 实例的主要进程负责与 ASM 相关的活动。

进程	说明
RBAL	在搜索过程中打开所有设备文件并协调重新平衡活动
ARBn	一个或多个从属进程，用于执行重新平衡活动
GMON	负责管理磁盘级活动，例如删除或脱机以及提高 ASM 磁盘组兼容性
MARK	根据需要将 ASM 分配单元标记为过时
Onnn	一个或多个 ASM 从属进程，形成与 ASM 实例的一组连接，用于交换消息
PZ9n	一个或多个并行从属进程，用于从 GV\$ 视图提取有关集群 ASM 安装的数据

版权所有 © 2010, Oracle。保留所有权利。

### ASM 组件：ASM 实例 – 主要进程

ASM 实例使用专用的后台进程完成其大部分功能。RBAL 进程在自动存储管理实例中协调磁盘组的重新平衡活动。它负责对自动存储管理磁盘执行全局打开操作。ARBn 进程在自动存储管理实例中执行实际的重新平衡数据区移动。可能同时存在多个这样的进程，这些进程名为 ARB0、ARB1 等等。GMON 进程维护 ASM 磁盘组中的磁盘成员资格。在向脱机磁盘写入失败后，MARK 进程将 ASM 分配单元标记为过时。Onnn 进程表示客户机/服务器连接的服务器端。启动实例时将出现这些进程，之后它们将消失。它们形成与 ASM 实例的一组连接，用于交换消息，仅在需要时才出现。PZ9n 进程表示一个或多个并行从属进程，当 ASM 同时在多台计算机上以集群配置运行时，可以使用该进程提取数据。

# ASM 实例初始化参数

ASM 实例使用 Oracle DB 实例所用参数的一小部分。

```
INSTANCE_TYPE = ASM
ASM_POWER_LIMIT = 1
ASM_DISKSTRING = '/dev/sda1','/dev/sdb*'
ASM_DISKGROUPS = DATA2, FRA
ASM_PREFERRED_READ_FAILURE_GROUPS = DATA.FailGroup2
DIAGNOSTIC_DEST = /u01/app/oracle
LARGE_POOL_SIZE = 12M
REMOTE_LOGIN_PASSWORDFILE = EXCLUSIVE
```

版权所有 © 2010, Oracle。保留所有权利。

## ASM 实例初始化参数

ASM 实例由参数文件控制，其方式与常规数据库实例相同。通常设置的参数包括：

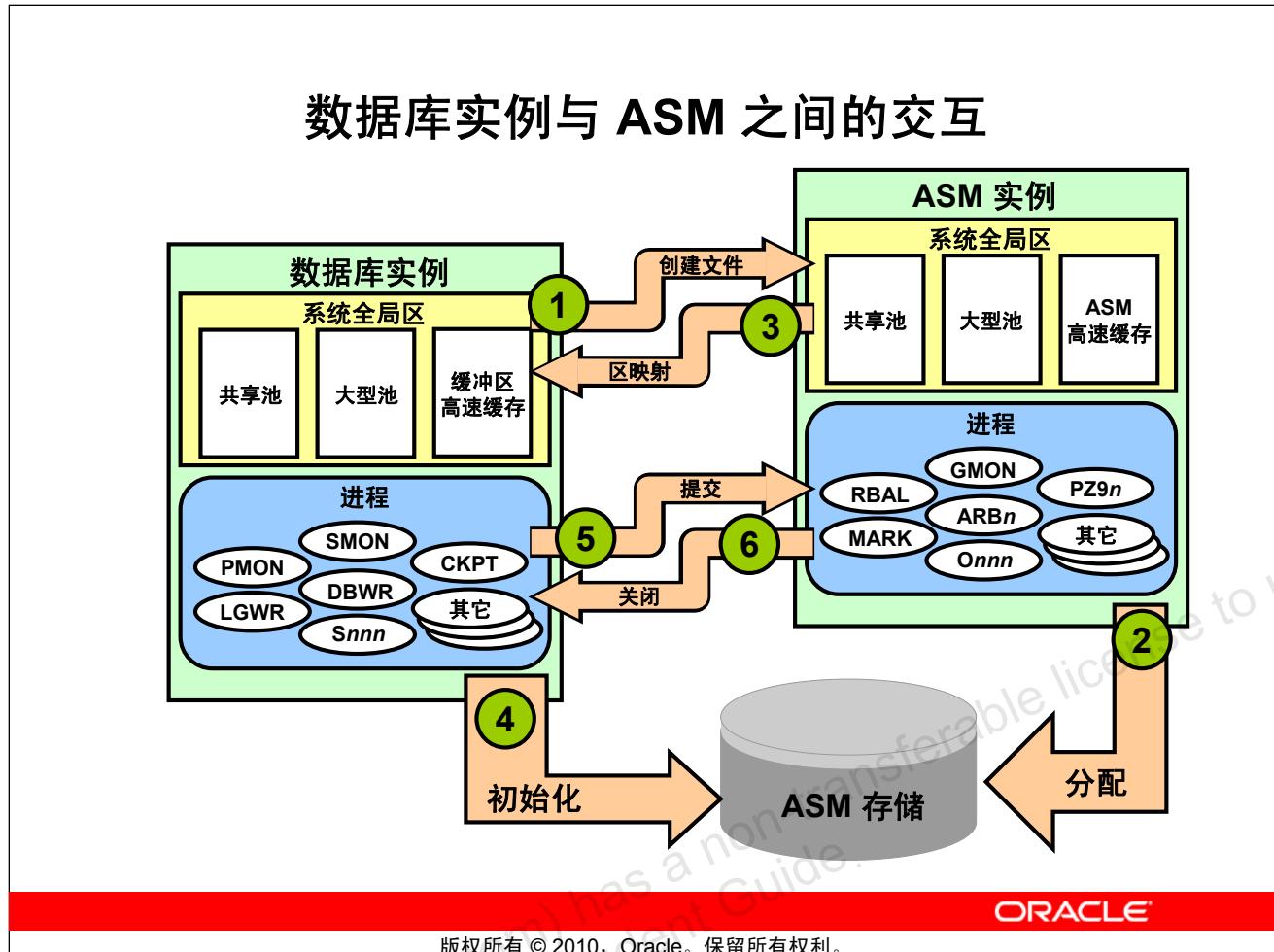
- INSTANCE\_TYPE 应该设置为 ASM，表示 ASM 实例。这是唯一一个必须定义的参数。对于数据库实例，该参数值设置为 RDBMS。
- ASM\_POWER\_LIMIT 控制重新平衡操作的速度。值的范围从 1 到 11，11 表示最快。如果省略，该值默认为 1。
- ASM\_DISKSTRING 是一个与操作系统相关的值，ASM 使用它来限制搜索时考虑的磁盘集。默认值为空字符串，这在大多数情况下就足够了。如上所示的约束性更强的值可以减少 ASM 执行搜索所需的时间，从而提高磁盘组装载次数。
- ASM\_DISKGROUPS 是 ASM 实例启动时或使用 ALTER DISKGROUP ALL MOUNT 命令时，ASM 实例要装载的磁盘组的名称列表。Oracle Restart 会装载列为相关磁盘组的磁盘组，即使这些磁盘组未与 ASM\_DISKGROUPS 参数列在一起也是如此。该参数没有默认值。
- ASM\_PREFERRED\_READ\_FAILURE\_GROUPS 指定包含首选读取磁盘的故障组。对于包含数据的镜像副本且有一个副本非常接近于服务器的扩展或延伸集群数据库，该参数非常有用。

## ASM 实例初始化参数（续）

- DIAGNOSTIC\_DEST 指定自动诊断资料档案库 (ADR) 主目录的位置。此目录下有跟踪文件、预警日志、核心文件和意外事件文件。此参数的默认值由 ORACLE\_BASE 的值派生。
- LARGE\_POOL\_SIZE 指定大型池分配堆的大小（以字节表示）。大型池分配堆用于共享服务器系统中的会话内存，供消息缓冲区的并行执行和磁盘 I/O 缓冲区的备份进程使用。ASM 实例使用自动内存管理，所以此参数用作大型池不能低于的最小大小。
- REMOTE\_LOGIN\_PASSWORDFILE 指定 Oracle 软件是否检查口令文件。默认值为 EXCLUSIVE。

上面列出的八个参数是需要为 ASM 实例创建的唯一几个非默认参数。ASM 实例与数据库实例不同，因为并不是所有的数据库参数都对 ASM 实例有效。在全部 344 个数据库实例参数中，大约有 74 个参数可以用于 ASM 实例。对于幻灯片中未列出的其它参数，虽然其默认值对于大多数安装来说应该已经足够了，但也可以根据需要进行设置。

**注：**ASM 实例默认启用自动内存管理，即使未明确设置 MEMORY\_TARGET 参数也是如此。该参数是全面 ASM 内存管理唯一需要设置的参数。Oracle Corporation 强烈建议您对 ASM 实例使用自动内存管理。



## 数据库实例与 ASM 之间的交互

文件创建过程可以很好地说明数据库实例与 ASM 之间发生的交互。文件创建过程如下所示：

1. 数据库请求创建文件。
2. ASM 前台进程创建一个持续操作目录 (COD) 条目并在磁盘组中为新的文件分配空间。
3. ASMB 数据库进程接收新文件的区映射。
4. 现在文件处于打开状态，数据库进程直接初始化该文件。
5. 初始化后，数据库进程请求提交文件创建。这会导致 ASM 前台进程清除 COD 条目并将文件标记为已创建。
6. 文件提交确认会隐式关闭该文件。将来发生 I/O 时，数据库实例需要重新打开该文件。

此示例强调了关于 ASM 体系结构的重要两点：

- 数据库实例和 ASM 实例协同工作。数据库实例必须与 ASM 交互，以便将数据库文件映射到 ASM 区。数据库实例还接收与 ASM 操作（例如磁盘组重新平衡）相关的持续消息流，这类操作可能锁定或移动 ASM 区。
- 数据库 I/O 不通过 ASM 实例来传输。实际上，数据库直接根据 ASM 文件执行 I/O 操作，如幻灯片中步骤 4 所示。

## ASM 实例：动态性能视图

ASM 实例托管基于内存的元数据表，这些表通过动态性能视图呈现。

- ASM 实用程序使用 SQL 语言访问这些表来检索仅含元数据的信息
- 包含许多专用的 ASM 相关视图，例如：

V\$ASM_ALIAS	V\$ASM_ATTRIBUTE	V\$ASM_CLIENT
V\$ASM_DISK	V\$ASM_DISK_IOSTAT	V\$ASM_DISK_STAT
V\$ASM_DISKGROUP	V\$ASM_DISKGROUP_STAT	V\$ASM_FILE
V\$ASM_OPERATION	V\$ASM_TEMPLATE	



版权所有 © 2010, Oracle。保留所有权利。

### ASM 实例：动态性能视图

对于任何实例而言，其主要功能之一就是存储基于内存的元数据表。这些表以前缀 X\$ 开头并且通常不进行记录。以前缀 V\$ 开头的一系列动态性能视图用于以定制形式显示 X\$ 内存表中包含的数据。这类信息以只读方式提供，仅具有权限的管理员可以访问。使用 SQL 语言从 ASM 检索这类信息。上面的幻灯片列出了包含 ASM 相关元数据的最常用动态性能视图。还有数百个其它动态性能视图，但是其中大多数视图都是空的，因为它们需要数据库实例装载数据库控制文件。ASM 实例不装载数据库控制文件。有关动态性能视图的完整列表，请参阅《Oracle Database Reference 11g Release 2 (11.2)》文档手册。

## ASM 系统权限

- ASM 实例没有数据字典，所以连接 ASM 只能使用下列系统权限。

ASM 权限	权限组（建议）	权限
SYSASM	OSASM (asmadmin)	全部管理权限
SYSDBA	OSDBA (asmdba)	可访问 ASM 中存储的数据，在当前版本中还具有 SYSASM 权限
SYSOPER	OSOPER (asmoper)	受限权限，可使用一组非破坏性 ALTER DISKGROUP 命令以及启动和停止 ASM 实例

- 创建 SYS 用户时，会自动为其分配 SYSASM 权限。

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### ASM 系统权限

ASM 实例没有数据字典，所以连接 ASM 实例只能使用以下三种系统权限之一：SYSASM、SYSDBA 或 SYSOPER。下面的列表讲述了这些 ASM 系统权限。

- **SYSASM:** 该权限提供对 ASM 实例的全部管理权限。
- **SYSDBA:** 该权限授予对 ASM 中存储的数据的访问权限，在当前版本中还授予 SYSASM 管理权限。
- **SYSOPER:** 该权限授予使用一组非破坏性 ALTER DISKGROUP 命令以及启动和停止 ASM 实例的权限。不允许使用 CREATE DISKGROUP 等其它命令。

安装 ASM 时，使用操作系统组对 SYSASM、SYSDBA 和 SYSOPER 权限进行验证。Oracle Universal Installer (OUI) 实用程序将这些组分别称为 OSASM、OSDBA 和 OSOPER 组，即 OSASM 组是授予了 SYSASM 权限的操作系统组。OSASM、OSDBA 和 OSOPER 组值的建议名称分别为 asmadmin、asmdba 和 asmoper。因此，SYSASM 是数据库使用的名称，OSASM 是 OUI 实用程序使用的名称，而 asmadmin 是操作系统使用的名称。它们都是指同一组用户。第一次创建 ASM 实例时，只定义 sys 和 asmsnmp 这两个 ASM 用户。

# 使用 Oracle Enterprise Manager 管理 ASM 用户

The screenshot shows the Oracle Enterprise Manager 11g interface for managing ASM users. It includes two main windows:

- Create User Window:** A modal dialog where a new user 'MFULLER' is being created with password '\*\*\*\*\*'. It lists 'Available Privileges' (SYSDBA, SYSOPER) and 'Granted Privileges' (SYSASM). Buttons include 'Show SQL', 'Cancel', and 'OK'.
- Edit User: SYS Window:** A modal dialog for editing the 'SYS' user. It shows the current password '\*\*\*\*\*' and allows changing it to 'MFULLER'. It also lists 'Available Privileges' (ASMSNMP, SYSDBA) and 'Granted Privileges' (SYSDBA, SYSOPER, SYSASM). Buttons include 'Show SQL', 'Revert', and 'Apply'.

Both windows have red arrows pointing to their respective 'Create' and 'Edit' buttons. The background shows the main Database Control interface with a table of users and a message about creating a password file for remote connection.

## 使用 Oracle Enterprise Manager 管理 ASM 用户

Oracle Enterprise Manager 允许您管理通过远程连接（使用口令文件验证）访问 ASM 的用户。这些用户是专门为 ASM 实例保留的。

仅当您作为 SYSASM 用户连接时才具有此功能。如果您作为 SYSDBA 或 SYSOPER 用户连接，该功能将隐藏。

- 单击“Create (创建)”按钮时将显示“Create User (创建用户)”页。
- 单击“Edit (编辑)”按钮时将显示“Edit User (编辑用户)”页。
- 通过单击“Delete (删除)”按钮可以删除创建的用户。

**注：**要以 SYSASM 角色登录 ASM，请单击该页顶部的“Preferences (首选项)”链接，然后单击“Preferred Credentials (首选身份证明)”链接。此时您将在目标类型列表中看到 ASM。单击 ASM 目标类型旁边的“Set Credentials (设置身份证明)”图标，定义具有 SYSASM 权限的帐户和口令。您需要从 Database Control 中注销并再次登录，更改才会生效。

## 启动和停止 ASM 实例使用 SQL\*Plus

使用 SQL\*Plus 启动和停止 ASM 实例与启动和停止数据库实例的方式相似。

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ sqlplus / AS SYSASM
SQL*Plus: Release 11.2.0.1.0 - Production on Wed Jul 8 20:46:46 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.
Connected to an idle instance.
SQL> startup
ASM instance started

Total System Global Area 284565504 bytes
Fixed Size                  1336028 bytes
Variable Size                258063652 bytes
ASM Cache                   25165824 bytes
ASM diskgroups mounted
ASM diskgroups volume enabled
SQL> shutdown abort
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 使用 SQL\*Plus 启动和停止 ASM 实例

在 SQL\*Plus 中使用 STARTUP 命令启动 ASM 实例，其方式与启动 Oracle DB 实例的方式相似。启动 ASM 实例时，请注意以下几点：

- 要使用 SQL\*Plus 连接 ASM 实例，请将 ORACLE\_SID 环境变量设置为 ASM SID。单实例数据的默认 ASM SID 为 +ASM，Oracle RAC 节点的默认 ASM SID 是 +ASM*node\_number*，其中 *node\_number* 是节点的编号。*oraenv* 脚本将设置 ORACLE\_BASE、ORACLE\_SID、ORACLE\_HOME 和 PATH 变量。
- 初始化参数文件必须包含下面的条目：  
INSTANCE\_TYPE = ASM  
该参数指示正在启动 ASM 实例，而不是数据库实例。
- 运行 STARTUP 命令时，该命令尝试装载初始化参数 ASM\_DISKGROUPS 指定的磁盘组，而不是尝试装载和打开某个数据库。如果尚未为 ASM\_DISKGROUPS 输入值，稍后可以通过 ALTER DISKGROUP...MOUNT 命令装载磁盘组。

## 使用 SQL\*Plus 启动和停止 ASM 实例（续）

下面的列表讲述了与 ASM 相关的 STARTUP 命令参数。

- **FORCE:** 在重新启动 ASM 实例之前向它发出 SHUTDOWN ABORT。
- **MOUNT 或 OPEN:** 装载 ASM\_DISKGROUPS 初始化参数中指定的磁盘组。这是未指定命令参数时的默认值。
- **NOMOUNT:** 启动 ASM 实例而不装载任何磁盘组。
- **RESTRICT:** 在受限模式下启动实例。RESTRICT 子句可以与 MOUNT、NOMOUNT 和 OPEN 子句组合使用。

在受限模式下，数据库实例不能使用磁盘组。即，数据库无法打开磁盘组中的文件。另外，如果磁盘组是由某个实例在受限模式下装载的，则该磁盘组无法由集群中的任何其它实例装载。通过受限模式，您可以在磁盘组上执行维护任务，而不会受到客户机的干扰。在磁盘组处于受限模式下时执行重新平衡操作，就不再需要在集群环境中的 ASM 实例之间传递锁定和取消锁定区映射消息。这样可以提高重新平衡操作的整体吞吐量。在维护期间结束时，您必须显式卸载该磁盘组并在正常模式下重新装载该磁盘组。

在 SQL\*Plus 中运行 SHUTDOWN 命令时将启动 ASM 关闭进程。在运行此命令之前，请确保已设置 ORACLE\_SID 和 ORACLE\_HOME 环境变量，从而您可以连接到 ASM 实例。

Oracle 强烈建议您在尝试关闭 ASM 实例之前关闭使用该 ASM 实例的所有数据库实例。

下面的列表讲述了与 ASM 相关的 SHUTDOWN 命令参数。

- **NORMAL:** ASM 在卸载所有磁盘组并关闭 ASM 实例之前，将等待任何正在进行的 SQL 完成执行。在关闭实例之前，ASM 将等待所有当前连接的用户从该实例断开。如果任何数据库实例与 ASM 实例连接，则 SHUTDOWN 命令将返回错误并保持 ASM 实例继续运行。NORMAL 是默认关闭模式。
- **IMMEDIATE 或 TRANSACTIONAL:** ASM 在卸载所有磁盘组并关闭 ASM 实例之前，将等待任何正在进行的 SQL 完成执行。ASM 不等待当前与实例连接的用户断开连接。如果任何数据库实例与 ASM 实例连接，则 SHUTDOWN 命令将返回错误并保持 ASM 实例继续运行。
- **ABORT:** ASM 实例立即关闭，而不按顺序卸载磁盘组。这将导致下次启动 ASM 时进行恢复操作。如果有任何数据库实例与 ASM 实例连接，则该数据库实例也将中止。

**注：**有连接的 RDBMS 实例时，不应用 NORMAL、IMMEDIATE 和 TRANSACTIONAL 形式的关闭，而将返回下面的错误：

```
ORA-15097: cannot SHUTDOWN ASM instance with connect
RDBMS instance
```

## 启动和停止 ASM 实例使用 `srvctl`

可以使用 Server Control 实用程序 (`srvctl`) 启动和停止 ASM 实例。

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for
  ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
    /u01/app/oracle
$ srvctl start asm -o mount
$ srvctl stop asm -f
```

可以使用 Server Control 实用程序 (`srvctl`) 检查 ASM 实例的状态。

```
$ srvctl status asm
ASM is running on edrsr25p1
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 使用 `srvctl` 启动和停止 ASM 实例

可以使用 Server Control 实用程序 (`srvctl`) 启动和停止 ASM 实例以及由 Grid Infrastructure 管理的其它资源。`srvctl` 实用程序位于 Grid Infrastructure 的 ORACLE\_HOME/bin 位置下和数据库安装的 ORACLE\_HOME/bin 位置下。管理 ASM、监听程序或 Oracle Restart 时，应该使用位于 Grid Infrastructure ORACLE\_HOME 中的 `srvctl` 实用程序。可以下列方式使用 `srvctl` 实用程序来控制 ASM:

- 启动 ASM 实例。
 

```
srvctl start asm [-o <start_option>]
      <start_option> 是有效的实例启动选项
      (FORCE、MOUNT、OPEN、NOMOUNT 或 RESTRICT) 之一 (可选)
```
- 停止 ASM 实例。
 

```
srvctl stop asm [-o <stop_option>] -f
      <stop_option> 是有效的实例关闭选项
      (NORMAL、IMMEDIATE、TRANSACTIONAL 或 ABORT) 之一 (可选) ,
      -f 是强制
```
- 报告 ASM 实例的状态。
 

```
srvctl status asm
```

## 启动和停止 ASM 实例使用 asmcmd

asmcmd 实用程序提供了一个访问 ASM 的命令行界面，无需使用 SQL 语言。

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ asmcmd
Connected to an idle instance.
ASMCMD> startup
ASM instance started

Total System Global Area 284565504 bytes
Fixed Size                 1336028 bytes
Variable Size              258063652 bytes
ASM Cache                  25165824 bytes
ASM diskgroups mounted
ASM diskgroups volume enabled
ASMCMD> shutdown --abort
ASM instance shut down
Connected to an idle instance.
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 使用 asmcmd 启动和停止 ASM 实例

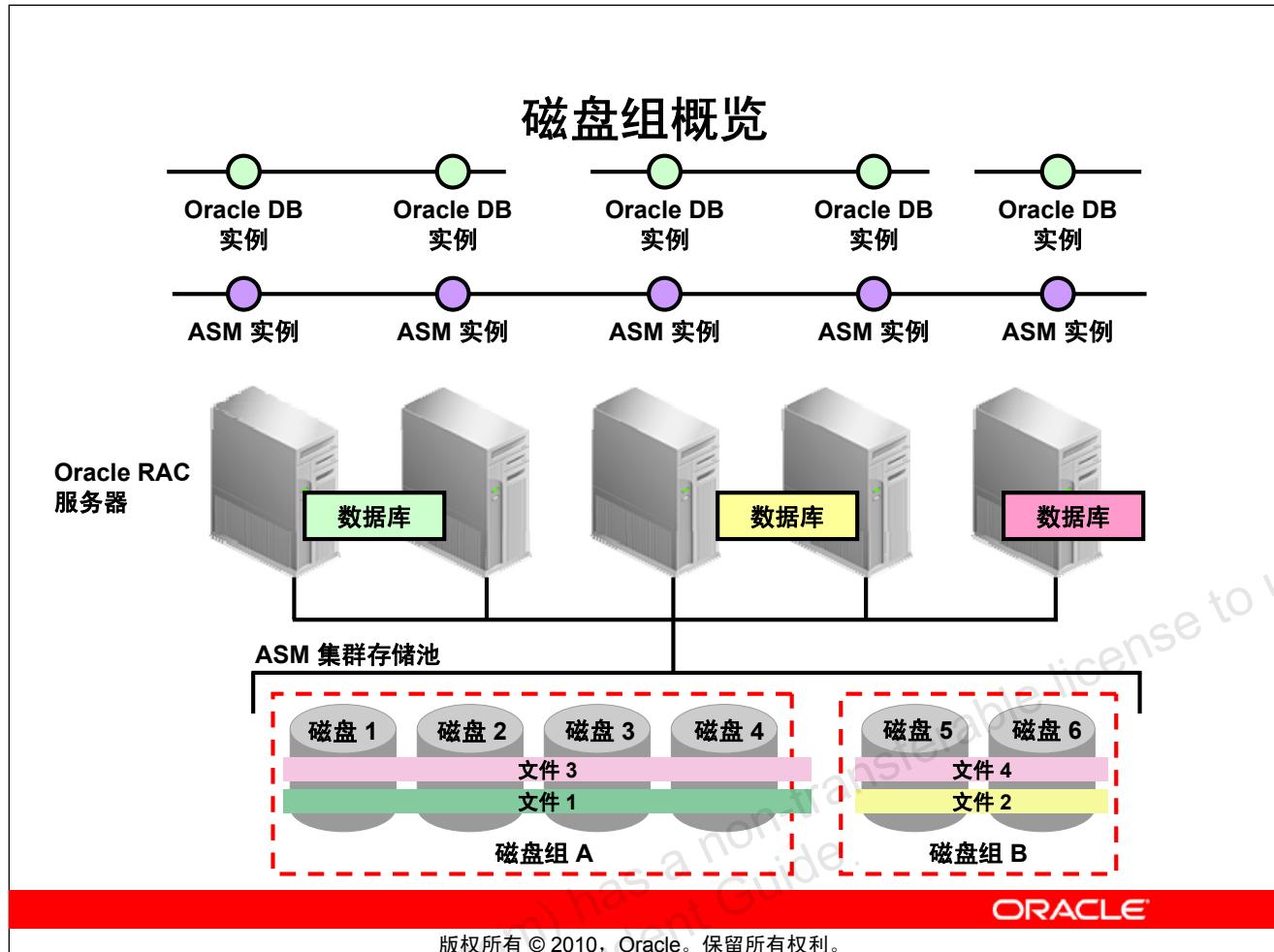
用于管理的 ASM 元数据位于 ASM 实例内包含的动态性能视图内。通常通过 ASM 实用程序使用 SQL 语言来访问这些视图。要求了解 SQL 语言增加了掌握 ASM 的学习时间，而且 SQL 语言培训通常不在系统管理员和存储管理员的学习范围内。asmcmd 实用程序提供了一个类似伪 Shell 的环境，可以接受 UNIX 格式的语法执行常用的 ASM 管理任务。它可以用于管理 Oracle ASM 实例、磁盘组、磁盘组的文件访问控制、磁盘组内的文件和目录、磁盘组的模板以及卷。

可以使用 asmcmd 实用程序启动和关闭 ASM 实例。支持的启动选项包括：

- nomount (在不装载磁盘组的情况下启动 ASM 实例)
- restrict (启动 ASM 实例并限制数据库使用)
- pfile <pfile.ora> (使用定制 pfile 启动 ASM 实例)

支持的关闭选项包括：

- immediate (立即执行关闭)
- abort (中止所有现有操作)



## 磁盘组概览

磁盘组是一个或多个磁盘的逻辑分组，ASM 将其作为一个集合来管理。每个磁盘组都包含与自身相关联的元数据。在概念上，ASM 磁盘组类似于典型存储区域网络中的逻辑卷。

从磁盘组内分配文件空间。磁盘组中存储的文件的内容将在该磁盘组的磁盘中平均分布或条带化，以便消除热点并在各磁盘上提供一致的性能。每个 ASM 文件都完全包含在一个磁盘组中。但是，一个磁盘组可以包含属于多个数据库的文件，而一个数据库可以使用来自多个磁盘组的不同文件。

磁盘组的关键属性是它的冗余设置。有三种可能的磁盘组冗余设置：

- 外部冗余，其中 ASM 不提供任何镜像，磁盘都假设为高度可靠。
- 正常冗余，其中 ASM 默认支持双向镜像，从而保证在较不可靠的存储上实现数据完整性。
- 高冗余，其中 ASM 默认支持三向镜像，从而更好地保证数据完整性。

ASM 最多支持创建 63 个磁盘组；但是，对于大多数安装，您很少会需要很多磁盘组。

# ASM 磁盘

## ASM 磁盘:

- 是为 ASM 磁盘组预配的存储设备
- 通过正常 O/S 接口来访问
- 必须可由 ASM 所有者读取和写入
- 必须可由集群中的所有节点访问
- 在不同的节点上可以有不同的 O/S 名称或路径
- 可以是:
  - 整个物理磁盘或物理磁盘的分区
  - 存储阵列中的磁盘或分区
  - 逻辑卷 (LV) 或逻辑单元 (LUN)
  - 网络连接文件 (NFS)

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## ASM 磁盘

ASM 磁盘组包含一个或多个 ASM 磁盘。

ASM 磁盘必须可由集群中所有节点的 ASM 所有者读取和写入。

ASM 磁盘可以是磁盘上的分区，但是 Oracle 强烈建议不要与其它应用程序共享 ASM 使用的物理磁盘。这是因为在应用 ASM 条带化和镜像策略时，磁盘组内的各个 ASM 磁盘的 I/O 性能必须相似且一致，才能获得最佳性能。在 ASM 和其它应用程序之间共享设备会难以确保磁盘性能相似且一致。

集群中不同节点上的 ASM 磁盘的 O/S 设备名称不必相同。ASM 通过读取 ASM 磁盘的标头来识别磁盘组的成员。

通常，ASM 磁盘是以存储阵列中的裸 LUN 提供给 ASM 的。另外，ASM 磁盘还可以是远程 NFS 文件器中的文件。

## 分配单元

ASM 磁盘划分为分配单元 (AU):

- 在创建磁盘组时可以配置 AU 大小。
- AU 默认大小为 1 MB:
  - 足够小，能够被数据库缓存；但又足够大，能够提供高效的后续访问
- 允许的 AU 大小:
  - 1、2、4、8、16、32 或 64 MB
  - 在超大型数据库 (VLDB) 中或使用专门的存储硬件时，较大的 AU 可能非常有用

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 分配单元

在 ASM 磁盘内，空间划分为分配单元 (AU)。AU 默认大小是 1 MB，该大小足够小，不会变为热点，但又足够大，能够提供高效的后续访问。创建磁盘组时可以设置 AU 大小。您无法更改磁盘组的 AU 大小。在超大型数据库 (VLDB) 中或使用专门的存储硬件时，较大的 AU 大小可能会很有用。如果经常访问某个 AU，则数据库内核会将其缓存以便实现更高的访问效率。

# ASM 文件

## ASM 文件:

- 是由 AU 组成的 ASM 区的集合
  - 可变大小的区支持大型文件
- 对数据库内核显示为普通文件
- 文件名以 “+” 开头
  - 例如,  
+DATA/orcl/datafile/system.256.689832921
- 可以与可选的文件名别名相关联
  - 例如, +DATA/dbfiles/mydb/system01.dbf
- 在磁盘组中的磁盘中平均分布
- 根据磁盘组中定义的策略进行镜像



版权所有 © 2010, Oracle。保留所有权利。

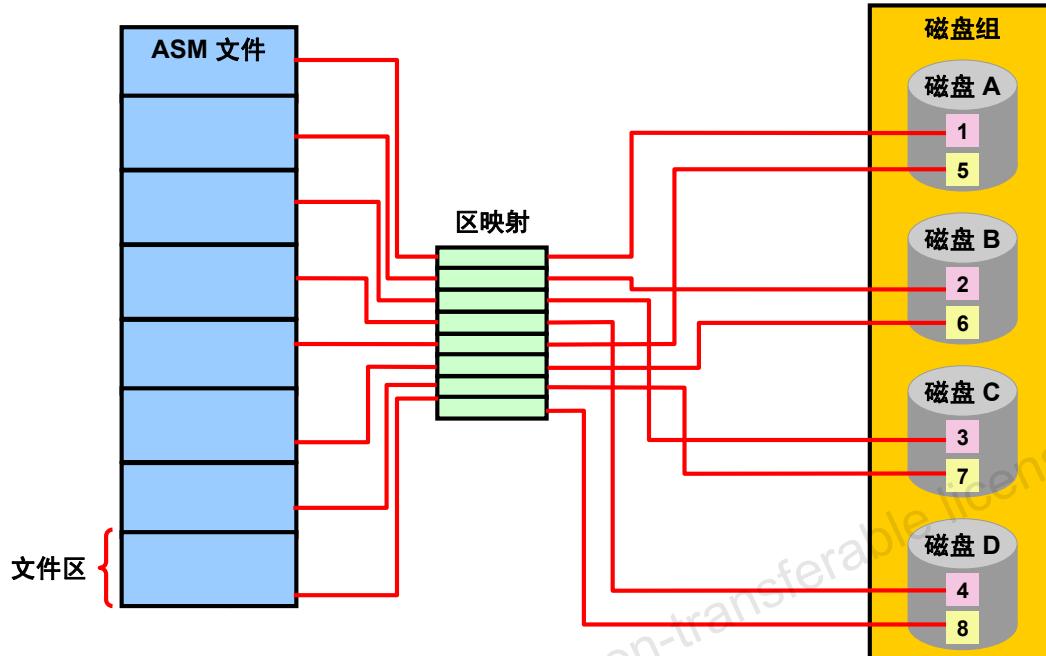
## ASM 文件

ASM 呈现一组文件供 ASM 的客户机使用。ASM 文件由一组分配单元组成，这些文件对数据库内核显示为普通文件。

每个 ASM 文件都具有唯一的系统生成的名称。幻灯片显示了一个完全限定的 ASM 文件名的示例。完全限定的 ASM 文件名表示一个层次结构，其开头是加号与磁盘组名称的组合。磁盘组名称后是数据库名称和文件类型。最后的元素由标记名称、文件编号和原型编号组成。可以创建别名（可选），为管理员提供一种更用户友好的方式来引用 ASM 文件。文件使用条带化和镜像所有内容 (SAME) 策略在磁盘组的各 ASM 磁盘中平均分布。

ASM 本身支持大多数与数据库相关的文件类型，例如数据文件、日志文件、控制文件、RMAN 备份及其它文件。在 Oracle Database 11g 发行版 2 之前，ASM 仅支持与 Oracle DB 相关的文件，无法用于存储和管理 ASCII 跟踪文件和预警日志、Oracle 二进制文件、Oracle 集群注册表 (OCR) 和集群表决磁盘。Oracle Database 11g 发行版 2 提供在 ASM 上运行通用文件系统的方式，从而消除了上述限制。

## 区映射



**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 区映射

ASM 通过称为区映射的元数据跟踪文件的分布。区映射是一个表，将文件中的数据区映射到磁盘上的分配单元。

文件区与分配单元之间的关系如下所示。区包含：

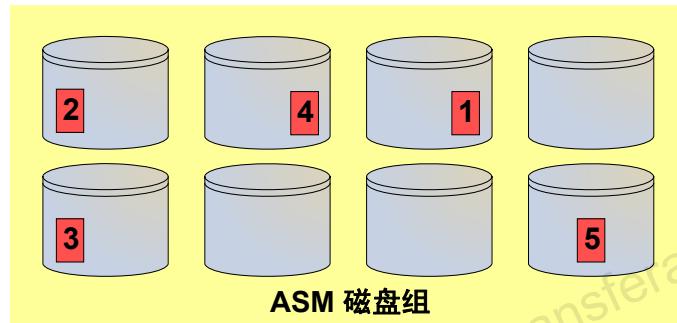
- 一个 AU 用于前 20,000 个区 (0-19999)
- 4 个 AU 用于接下来的 20,000 个区 (20000-39999)
- 16 个 AU 用于 40,000 以上的区

将可变大小的区与大型分配单元结合使用，可满足非常大的 ASM 文件的需要。

## 条带化粒度

ASM 用于负载平衡的条带化策略与用于等待时间的条带化策略不同：

- 对于负载平衡，粗粒度条带化将多个分配单元连接起来使用。
  - 例如：



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 条带化粒度

在 ASM 中，条带化主要具有两种用途：

- 在磁盘组的所有磁盘中执行 I/O 负载平衡。
- 改善 I/O 等待时间。

粗粒度条带化在磁盘组的各磁盘中分布分配单元。这用于为磁盘组提供负载平衡。分配文件时，ASM 会在所有磁盘中平均分布分配单元。有时分布无法非常平均，但是随着时间的流逝，会变得几乎相等。上图显示了一个具有五个分配单元的文件，这些分配单元在条带化之后，分布在一个外部冗余磁盘组的五个磁盘中，该磁盘组共包含八个磁盘。

对于前 20,000 个区，区大小等于 AU 大小。从第 20,001 个区到第 40,000 个区，该区组总是一次分配 8 个区，区大小等于 4 倍 AU 大小。如果 AU 大小为 1 MB，这意味着 ASM 文件将一次增大 64 MB ( $8 * 4 * 1 \text{ MB}$ )。如果文件采用粗粒度条带化，则它将在 8 个区中进行条带化，条带大小为 1 AU。条带化始终在 AU 级别进行，而不是在区级别。因此，无论文件多大，粗粒度文件的每个 AU 都与该文件的前一个 AU 所在的磁盘不同。在第 40,000 个区之后，仍旧是每次分配 8 个区，但区大小等于 16 倍 AU 大小。

## 细粒度条带化

细粒度条带化在分配单元组中放置 128 KB 条带单元来改善等待时间。

- 磁盘组具有 8 个磁盘和外部冗余
- 使用默认的 AU 大小：1 MB
- 第一个 1 MB 区在 8 个 AU 中以 128 KB 条带形式写入



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 细粒度条带化

细粒度条带化将数据区拆分为 128 KB 的块，然后在许多磁盘中分布每个区的负载，从而改善特定类型文件的等待时间。默认情况下，细粒度条带化用于控制文件和联机重做日志文件。

本页的图显示了细粒度条带化的工作原理。在此示例中，新文件的第一个 1 MB 区最终占用 8 个不同分配单元中的第一个 128 KB 块，这 8 个分配单元分布在磁盘组的八个磁盘中。因此，1 MB 的读取或写入分布在八个磁盘（而不是一个磁盘）中。

## 细粒度条带化

示例：

- 磁盘组具有 8 个磁盘和外部冗余
- 使用默认的 AU 大小：1 MB
- 下一个 1 MB 区以 128 KB 条带的形式在相同的 8 个分配单元中写入，直到这些分配单元写满



ORACLE

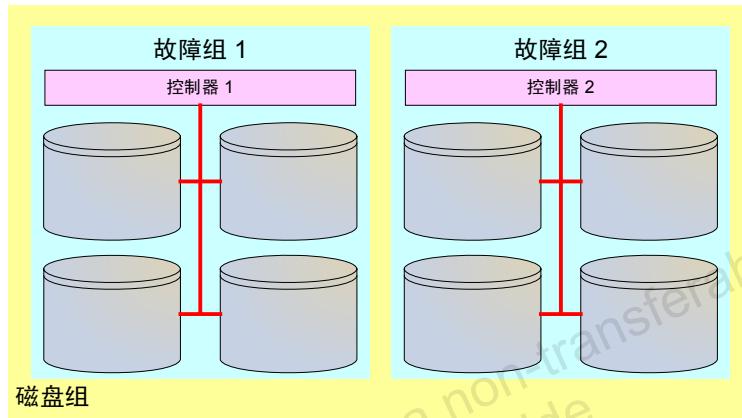
版权所有 © 2010, Oracle。保留所有权利。

### 细粒度条带化（续）

继续前面的示例，空间中的下一个 1 MB 区将分布在相同分配单元组的每个单元的第二个 128 KB 块中。这种模式将继续，直到第一组分配单元写满，然后分配另一组。

## ASM 故障组

- 共享需要容错的一项公用资源的一组磁盘
- 镜像的区副本存储在不同的故障组中
- 存储硬件指示故障组边界
  - 分离磁盘控制器的示例:



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### ASM 故障组

在磁盘组内，可以将一些磁盘归到故障组中。故障组是存储或数据库管理员指定 ASM 镜像操作的硬件边界的一种方式。

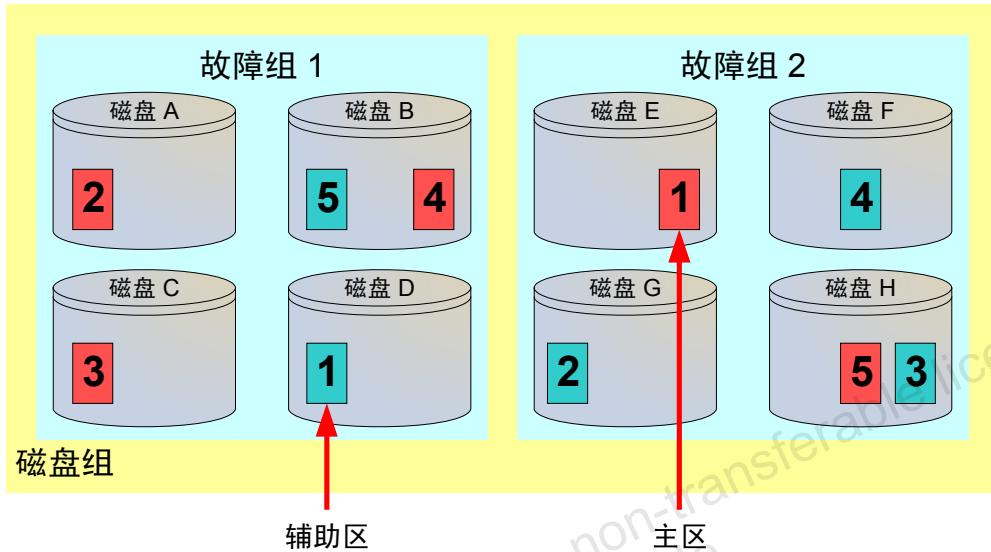
例如，可以将连接到一个磁盘控制器的所有磁盘指定为某个公用故障组的磁盘。这样，将在连接到不同控制器的磁盘上镜像文件区。另外，管理员可以配置 ASM 来选择默认故障组策略。默认策略是每个磁盘位于自己的故障组中。

您可以使用所需的任何标准将磁盘归到故障组中。故障组可以用于避免单个磁盘、磁盘控制器、I/O 网络组件以及甚至整个存储系统的故障。通常，管理员将分析其存储环境并对故障组进行组织来消除特定故障情况。

由数据库或存储管理员确定其安装的最佳故障组配置。

## 条带化和镜像示例

一个共含八个磁盘的正常冗余磁盘组，其中的磁盘分布在两个故障组中。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

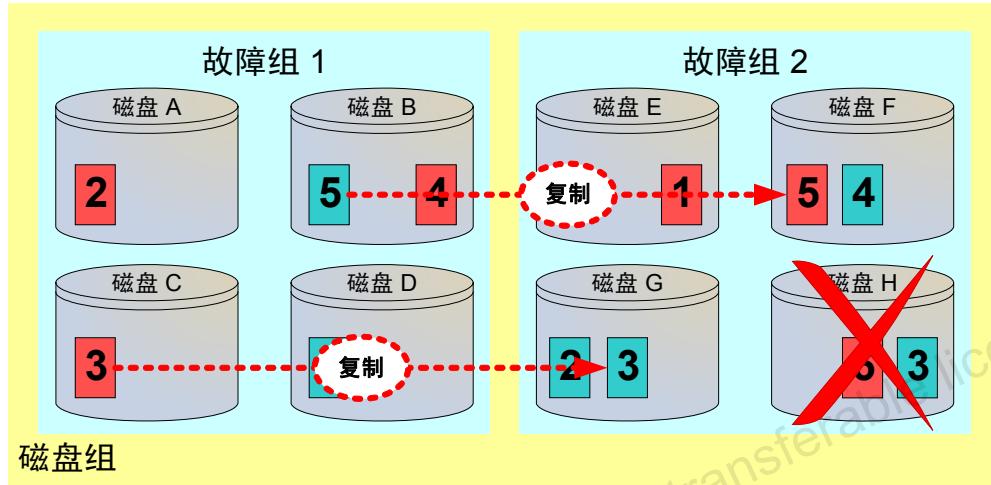
### 条带化和镜像示例

本页中的图显示了一个正常冗余磁盘组中条带化和镜像的示例。红块表示一个包含五个区的文件，这些区在条带化后，分布在磁盘组（共八个磁盘）的五个磁盘中。蓝块表示文件区的镜像副本。请注意，无论区在各个磁盘和故障组中如何分布，每个区在一个故障组中仅有一个副本。

分配文件时，主区的分配用于保证性能，辅助副本的分配用于保证完整性。因此，默认情况下，所有数据库读取都针对主区进行。

## 故障示例

如果磁盘 H 发生故障，则将根据正常镜像在正常磁盘上重新创建该磁盘包含的区。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 故障示例

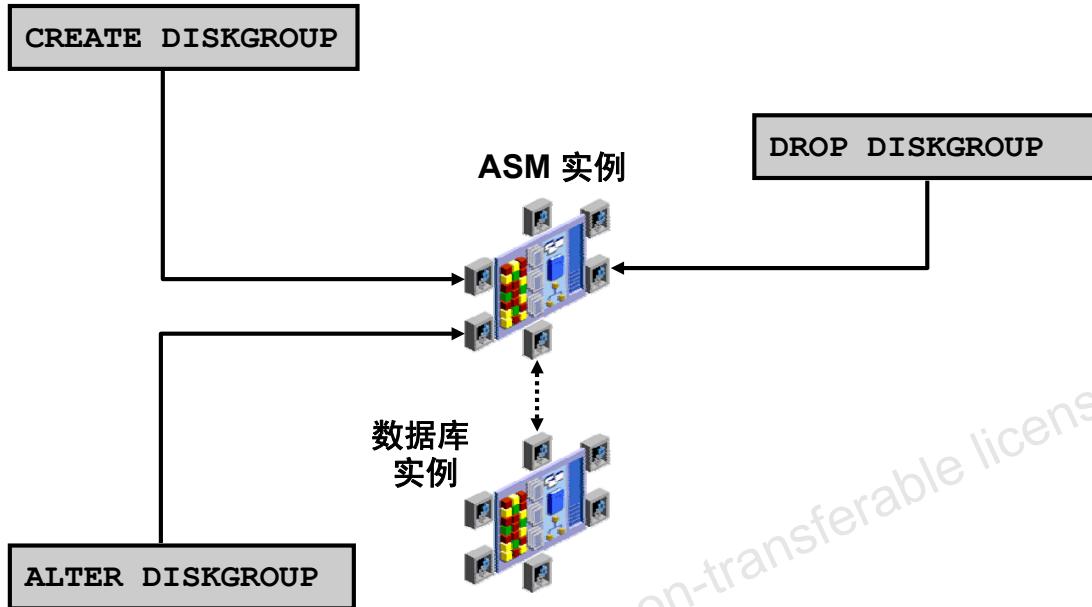
此处扩展了上一页中的示例，假设磁盘 H 发生故障，无法再访问其包含的数据。该故障要求恢复故障磁盘上的所有区并将这些区复制到另一磁盘。

区 3 和 5 的正常副本将复制到同一故障组中另一磁盘的空闲区域。在此示例中，区 5 将从磁盘 B 复制到磁盘 F，区 3 将从磁盘 C 复制到磁盘 G。

磁盘发生故障后的最后一步是 ASM 从磁盘组中删除故障磁盘。

删除磁盘基本上会触发相同的过程；不过，在删除磁盘时，要被删除的磁盘上的区首先会复制到可用替代位置。

## 管理磁盘组



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 管理磁盘组

ASM 实例的主要目的是管理磁盘组和保护其数据。ASM 实例还将文件分布传送到数据库实例。这样，数据库实例就可以直接访问磁盘组中存储的文件。

有多个磁盘组管理命令。执行这些命令都需要 SYSASM 或 SYSDBA 权限，且必须从 ASM 实例发出。

您可以添加新磁盘组。还可以修改现有磁盘组来添加新磁盘、删除现有磁盘以及执行许多其它操作。您可以删除现有磁盘组。

## 创建和删除磁盘组使用 SQL\*Plus

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
The Oracle base for ORACLE_HOME=/u01/app/oracle/product/11.2.0/grid is
/u01/app/oracle
$ sqlplus / AS SYSASM
SQL*Plus: Release 11.2.0.1.0 - Production on Wed Jul 8 20:46:46 2009
Copyright (c) 1982, 2009, Oracle. All rights reserved.

..
SQL> CREATE DISKGROUP dgroupA NORMAL REDUNDANCY
FAILGROUP controller1 DISK
  '/devices/A1' NAME diskA1 SIZE 120G FORCE,
  '/devices/A2',
FAILGROUP controller2 DISK
  '/devices/B1',
  '/devices/B2';


```

```
SQL> DROP DISKGROUP dgroupA INCLUDING CONTENTS;
```

版权所有 © 2010, Oracle。保留所有权利。

### 创建和删除磁盘组

假设 ASM 磁盘搜索在 /dev 目录中找到了下列磁盘：A1、A2、B1 和 B2。另外，假设磁盘 A1 和 A2 的磁盘控制器不同于 B1 和 B2 的磁盘控制器。幻灯片中的第一个示例说明如何配置名为 DGROUPA 的磁盘组，其包含两个故障组：CONTROLLER1 和 CONTROLLER2。该示例还对该磁盘组使用默认冗余属性 NORMAL\_REDUNDANCY。您还可以提供磁盘名称和磁盘的大小（可选）。如果您不提供此信息，ASM 会创建默认名称并尝试确定磁盘的大小。如果无法确定大小，将返回错误。FORCE 指示指定的磁盘应该添加到指定的磁盘组，即使该磁盘已经格式化为某个 ASM 磁盘组的成员。对未格式化为某个 ASM 磁盘组的成员的磁盘使用 FORCE 选项将返回错误。

如幻灯片中第二个语句所示，您可以删除磁盘组及其所有文件。为了避免意外删除，如果磁盘组除内部 ASM 元数据外还包含任何其它文件，则必须指定 INCLUDING CONTENTS 选项。必须先装载磁盘组才能将其删除。在确保未打开任何磁盘组文件后，将从磁盘组中删除该组及其所有驱动器。然后，将覆盖每个磁盘的标头来清除 ASM 格式信息。

## 向磁盘组添加磁盘

```
ALTER DISKGROUP dgroupA ADD DISK
  '/dev/sde1' NAME A5,
  '/dev/sdf1' NAME A6,
  '/dev/sdg1' NAME A7,
  '/dev/sdh1' NAME A8;
```

```
ALTER DISKGROUP dgroupA ADD DISK '/devices/A*' ;
```

磁盘格式化

磁盘组重新平衡

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 向磁盘组添加磁盘

此示例显示如何向磁盘组添加磁盘。可以执行 ALTER DISKGROUP ADD DISK 命令来添加磁盘。第一个语句向 DGROUPA 磁盘组添加四个新磁盘。

第二个语句说明搜索字符串的相互影响。请考虑下列配置：

/devices/A1 是磁盘组 DGROUPA 的成员。

/devices/A2 是磁盘组 DGROUPA 的成员。

/devices/A3 是磁盘组 DGROUPA 的成员。

/devices/A4 是候选磁盘。

第二个命令将 A4 添加到 DGROUPA 磁盘组。该命令将忽略其它磁盘，即使它们与搜索字符串相匹配，因为它们已经是 DGROUPA 磁盘组的一部分。上图显示当您向磁盘组添加磁盘时，ASM 实例会确保该磁盘是可寻址且可用的。然后会对该磁盘进行格式化和重新平衡。重新平衡过程比较耗费时间，因为该过程会将所有文件的区移至新磁盘。

**注：**重新平衡不会阻止任何数据库操作。重新平衡过程主要影响系统的 I/O 负载。重新平衡的能力越高，对系统施加的 I/O 负载越多，可用于数据库 I/O 的 I/O 带宽也就越少。

## 其它 ALTER 命令

从 dgroupA 删除磁盘:

```
ALTER DISKGROUP dgroupA DROP DISK A5;
```

在单个命令中添加和删除磁盘:

```
ALTER DISKGROUP dgroupA
DROP DISK A6
ADD FAILGROUP controller3
DISK '/dev/sd1' NAME A9;
```

取消磁盘删除操作:

```
ALTER DISKGROUP dgroupA UNDROP DISKS;
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 其它 ALTER 命令

幻灯片中的第一个语句显示如何从 DGROUPA 磁盘组中删除一个磁盘。第二个语句显示如何在单个命令中添加和删除磁盘。这种方式的一大优点就是在该命令完成后才开始重新平衡操作。第三个语句显示如何取消磁盘删除操作。UNDROP 命令仅对磁盘的暂挂删除操作起作用；它对于已经完成的删除操作没有影响。

如果需要，可以使用下面的语句对 DGROUPB 磁盘组进行重新平衡:

```
ALTER DISKGROUP dgroupB REBALANCE POWER 5;
```

此命令通常是不需要的，因为在添加、删除磁盘或调整磁盘大小时会自动执行该操作。但是如果您希望使用 POWER 子句覆盖由初始化参数 ASM\_POWER\_LIMIT 定义的默认速度，此命令非常有用。通过重新输入具有新级别的命令，可以更改正在进行的重新平衡操作的能力级别。能力级别为零会导致重新平衡操作中断，直到隐式或显式重新调用该命令。下面的语句可以卸载 DGROUPA:

```
ALTER DISKGROUP dgroupA DISMOUNT;
```

通过 MOUNT 和 DISMOUNT 选项，您可以使一个或多个磁盘组对数据库实例可用或不可用。在支持单实例的集群 ASM 环境中，当该实例故障转移到另一节点时，手动卸载和装载功能非常有用。

# 使用 Oracle Enterprise Manager 管理 ASM

**General**

Current Status: Up  
Up Since: Jul 8, 2009 10:18:28 AM GMT+07:00  
Availability (%): 76.38 (Last 24 hours)  
Instance Name: +ASM  
Version: 11.2.0.1.0  
Host: edrsr25p1.us.oracle.com  
Oracle Home: /u01/app/oracle/product/11.2.0/grid

**Disk Group Usage (GB)**

Disk Group	Size (GB)	Type
FRA	~9.0	Unallocated
DATA	~4.5 (Used ~3.97 GB)	Internal

**Diagnostic Summary**

Alert Log: No ORA- errors  
Active Incidents: 0

**Serviced Databases**

Name	Disk Groups	Failure Groups	Allocated Space (GB)	Availability	Alerts
orcl.example.com	FRA, DATA	4 (0 down)	3.97	1	1 0
CLUSTER_UNKNOWN	DATA	4 (0 down)	0	Not Monitored	

**Serviced ASM Cluster File Systems**

Mount Point	Availability	State	Used (%)	Used (GB)	Size (GB)	Allocated Space (GB)	Volume	Disk Group
(No ASM Cluster File Systems)								

ORACLE

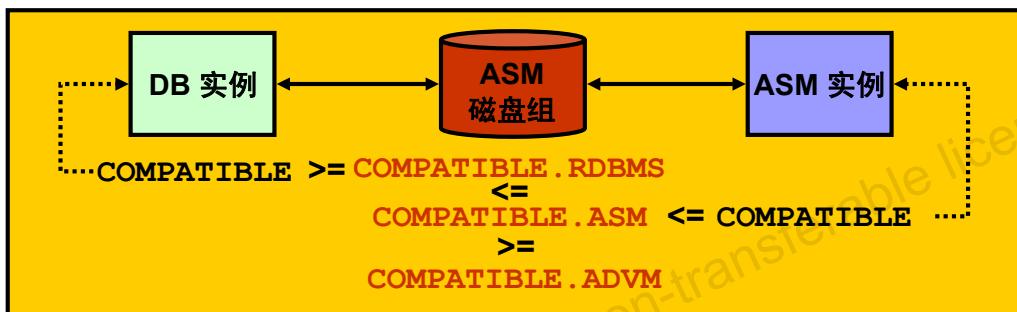
版权所有 © 2010, Oracle。保留所有权利。

## 使用 Oracle Enterprise Manager 管理 ASM

Oracle Enterprise Manager (EM) 是 Oracle 的管理工具系列，管理对象涵盖数据库、中间件、应用程序、网络、IT 基础结构及其它。Oracle Enterprise Manager 是一个基于浏览器的环境，为常见 ASM 管理任务提供了一个点击式替代方案。

## ASM 磁盘组兼容性

- 每个磁盘组的兼容性都可以分别控制：
  - ASM 兼容性控制磁盘上 ASM 元数据结构。
  - RDBMS 兼容性控制最低使用者客户机级别。
  - ADVM 兼容性确定磁盘组是否可以包含 Oracle ASM 卷。
- 设置磁盘组兼容性的操作是不可逆的。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### ASM 磁盘组兼容性

适用于 ASM 磁盘组的兼容性有三种：涉及描述磁盘组的持久性数据结构，客户机（磁盘组的使用者）的功能，以及能否在磁盘组中包含卷。这些属性分别称为“ASM 兼容性”、“RDBMS 兼容性”和“ADVM 兼容性”。每个磁盘组的兼容性都可以独立控制。这是支持包含 Oracle Database 10g 和 Oracle Database 11g 磁盘组的异构环境所必需的。这三种兼容性设置是每个 ASM 磁盘组的属性：

- RDBMS 兼容性是指 RDBMS 实例的最低兼容版本，此设置将允许该实例装载磁盘组。该兼容性确定了 ASM 实例与数据库 (RDBMS) 实例间交换消息的格式。ASM 实例可以支持以不同兼容性设置运行的不同 RDBMS 客户机。每个实例的数据库兼容版本设置必须高于或等于该数据库使用的所有磁盘组的 RDBMS 兼容性。数据库实例与 ASM 实例通常在不同的 Oracle 主目录中运行。这意味着数据库实例可以运行与 ASM 实例不同的软件版本。数据库实例第一次连接到 ASM 实例时，系统会协定这两个实例都支持的最高版本。

数据库的兼容性参数设置、数据库的软件版本以及磁盘组的 RDBMS 兼容性设置确定了数据库实例能否装载给定的磁盘组。

## ASM 磁盘组兼容性（续）

- ASM 兼容性是指控制磁盘上 ASM 元数据的数据结构格式的持久性兼容性设置。磁盘组的 ASM 兼容性级别必须始终高于或等于同一磁盘组的 RDBMS 兼容性级别。ASM 兼容性只与 ASM 元数据的格式相关。文件内容的格式取决于数据库实例。例如，可以将某个磁盘组的 ASM 兼容性设置为 11.0，而将该磁盘组的 RDBMS 兼容性设置为 10.1。这意味着该磁盘组只能由软件版本为 11.0 或更高的 ASM 软件管理，而软件版本高于或等于 10.1 的任何数据库客户机都可以使用该磁盘组。
- ADVM 兼容性属性确定磁盘组能否包含 Oracle ASM 卷。该值必须设置为 11.2 或更高。设置该属性前，必须确保 COMPATIBLE.ASM 的值为 11.2 或更高。此外，还必须加载 ADVM 卷驱动程序。

仅当持久性磁盘结构或消息传送协议发生更改时，才需要提高磁盘组的兼容性。但是，提高磁盘组兼容性是一个不可逆的操作。可以使用 CREATE DISKGROUP 命令或 ALTER DISKGROUP 命令来设置磁盘组兼容性。

**注：**除了磁盘组兼容性，兼容参数（数据库兼容版本）确定了启用的功能；该参数适用于数据库或 ASM 实例，具体取决于 *instance\_type* 参数。例如，将该参数设置为 10.1 将禁止使用 Oracle Database 11g 中引入的任何新功能（磁盘联机/脱机、可变区等）。

## ASM 磁盘组属性

名称	属性	值	说明
au_size	创建、更改	1 2 4 8 16 32 64MB	磁盘组中分配单元的大小
compatible.rdbms	创建、更改	有效的数据库版本	数据库与 ASM 之间交换的消息的格式
compatible.asm	创建、更改	有效的 ASM 实例版本	磁盘上 ASM 元数据结构的格式
compatible.advm	创建、更改	有效的 ASM 实例版本	允许在磁盘组中包含 Oracle ASM 卷
disk_repair_time	创建、更改	0 M 到 $2^{32}$ D	磁盘脱机之后删除此磁盘之前的时间量
template.tname. redundancy	更改	UNPROTECT MIRROR  HIGH	指定模板的冗余
template.tname. stripe	更改	COARSE FINE	指定模板的条带化属性

```
CREATE DISKGROUP DATA2 NORMAL REDUNDANCY
DISK '/dev/sda1','/dev/sdb1'
ATTRIBUTE 'compatible.asm'='11.2';
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### ASM 磁盘组属性

每当创建或更改 ASM 磁盘组时，都可以使用 CREATE DISKGROUP 和 ALTER DISKGROUP 命令新增的 ATTRIBUTE 子句更改其属性。幻灯片中的表简要概述了这些属性：

- ASM 允许使用在创建磁盘组时指定的不同分配单元 (AU) 大小。AU 大小可以是 1、2、4、8、16、32 或 64 MB。
- RDBMS 兼容性：有关详细信息，请参阅“ASM 磁盘组兼容性”一节。
- ASM 兼容性：有关详细信息，请参阅“ASM 磁盘组兼容性”一节。
- 可以用分钟 (M)、小时 (H) 或天 (D) 为单位指定 DISK\_REPAIR\_TIME。如果省略单位，则使用默认值 H。如果省略此属性，则使用默认值 3.6H。可以使用 ALTER DISKGROUP ... DISK OFFLINE 语句覆盖此属性。
- 还可以为指定的模板指定冗余属性。
- 还可以为指定的模板指定条带化属性。

**注：**对于每个已定义的磁盘组，可以通过 V\$ASM\_ATTRIBUTE 固定视图查看所有已定义的属性。

## 使用 Oracle Enterprise Manager 编辑磁盘组属性

**ORACLE Enterprise Manager 11g**

Database Control

Automatic Storage Management +ASM edrsr25p1.us.oracle.com > Logged in As SYS / SYSASM

Disk Group: DATA

General Performance Templates Files Access Control Volumes

General

Name	DATA
State	MOUNTED
Redundancy	NORMAL
Total Size (GB)	9
Pending Operations	0
Allocation Unit (MB)	1

Advanced Attributes

Database Compatibility	10.1.0.0.0
ASM Compatibility	11.2.0.0.0
ASM Volume Compatibility	
Disk Repair Time (Hours)	3.6
Smart Scan Capability	Disabled
File Access Control	Disabled

**Edit Advanced Attributes for Disk Group: DATA**

**Disk Group Compatibility**

Advancing the disk group compatibility enables the user to use new features available in the newer version. This operation can not be reversed.

Database Compatibility	10.1.0.0.0	The minimum software version required for a database instance to use files in this disk group (10.1 and above).
ASM Compatibility	11.2.0.0.0	The minimum software version required for an ASM instance to mount this disk group (10.1 and above).
ASM Volume Compatibility		The minimum software version required for an ASM Volume to use this disk group (11.2 and above).

TIP The database compatibility has to be less than or equal to the ASM compatibility. The ASM Volume compatibility can only be set when ASM compatibility is 11.2 and above.

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 使用 Oracle Enterprise Manager 编辑磁盘组属性

Oracle Enterprise Manager 提供了一种简便方式用于存储和检索与磁盘组相关的环境设置。

在“Create Disk Group（创建磁盘组）”页和“Edit Advanced Attributes for Disk Group（编辑磁盘组的高级属性）”页中都可以设置兼容属性。仅“Edit Advanced Attributes for Disk Group（编辑磁盘组的高级属性）”页中添加了 `disk_repair_time` 属性。

注：对于低于 11g 的 ASM 实例，默认 ASM 兼容性和客户机兼容性都为 10.1。对于 11g ASM 实例，默认 ASM 兼容性为 11.2，数据库兼容性为 10.1。

## 检索 ASM 元数据

- 使用 SQL\*Plus:

```
SQL> SELECT f.type, f.redundancy, f.striped, f.modification_date,
  a.system_created, a.name FROM v$asm_alias a, v$asm_file f WHERE
  a.file_number = f.file_number and a.group_number = f.group_number
  and type='DATAFILE';
  TYPE      REDUND    STRIPE   MODIFICAT   S   NAME
  -----  -----  -----  -----  -  -----
  DATAFILE  MIRROR    COARSE   08-JUL-09   Y  SYSTEM.256.689832921
  DATAFILE  MIRROR    COARSE   08-JUL-09   Y  SYSAUX.257.689832923
  ..
```

- 使用 asmcmd:

```
ASMCMD> ls -l +DATA/orcl/datafile
Type      Redund    Striped   Time                  Sys   Name
DATAFILE  MIRROR    COARSE   JUL 08 21:00:00   Y  SYSTEM.256.689832921
DATAFILE  MIRROR    COARSE   JUL 08 21:00:00   Y  SYSAUX.257.689832923
..
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

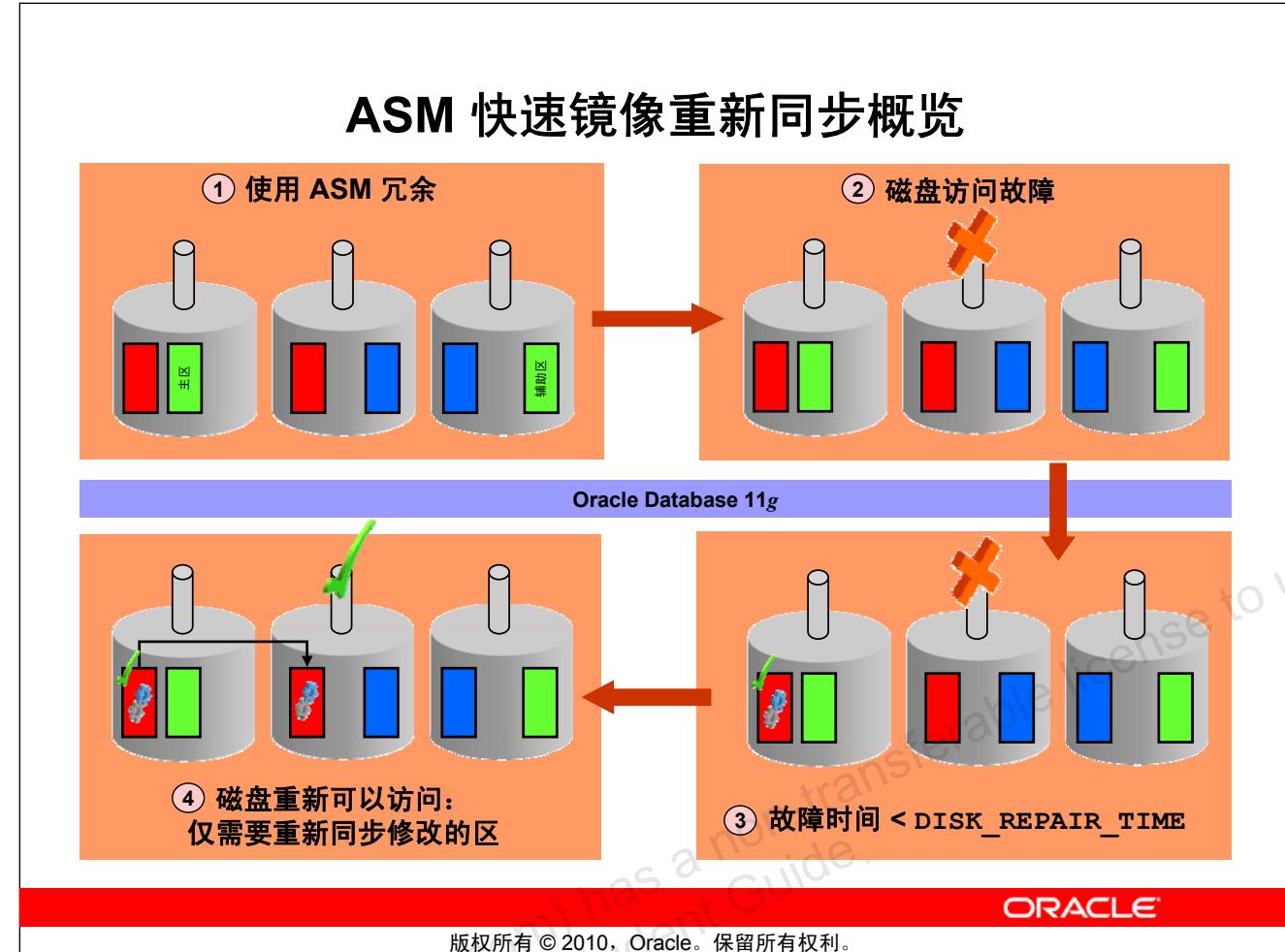
## 检索 ASM 元数据

ASM 实例托管基于内存的元数据表，这些表通过动态性能视图呈现。可以使用 SQL\*Plus、asmcmd 或 Oracle Enterprise Manager 查询该数据。

使用 SQL\*Plus 需要具有 SQL 语言知识，可能要联接多个动态性能视图来检索相关信息。幻灯片中的第一个示例显示 v\$asm\_file 和 v\$asm\_alias 之间建立了一个联接，以便显示关于数据库的数据文件的元数据。如果针对 ASM 实例执行此查询，根据语法的定义方式，将检索多个数据库的数据文件。需要使用附加过滤条件才能将输出限定为单个数据库。

asmcmd 实用程序的优点在于，不需要 SQL 语言知识就能连接到 ASM 实例并检索元数据。它使用与 UNIX 表示法类似的样式。此幻灯片上的第二个示例使用 asmcmd 命令来检索 SQL 示例中的元数据。此示例的另一个优点是输出限制为单个数据库的数据文件，因为列出的路径包含数据库名称 orcl 和文件类型 datafile。因此，对于 asmcmd 中显示的目录，需要使用 WHERE 子句设置 SQL 过滤条件才能给出相同结果。

**注：**在 Oracle Enterprise Manager Database Control 中，只需在各个 ASM Web 页中浏览就能查看大多数 ASM 元数据。



## ASM 快速镜像重新同步概览

ASM 快速镜像重新同步可以显著减少重新同步临时故障磁盘所需的时间。如果某个磁盘因临时故障而脱机，ASM 将跟踪在中断期间发生修改的区。临时故障被修复后，ASM 可以快速地仅重新同步在中断期间受影响的 ASM 磁盘区。

此功能假设受影响的 ASM 磁盘的内容未被破坏或修改。

某个 ASM 磁盘路径出现故障时，如果您已设置了相应磁盘组的 DISK\_REPAIR\_TIME 属性，则 ASM 磁盘会脱机，但不会被删除。此属性的设置确定了 ASM 可容忍的磁盘中断持续时间；如果中断在此时间范围内，则修复完成后仍可重新同步。

注：跟踪机制对每个修改的分配单元使用 1 位。这样可确保跟踪机制非常高效。

## 小测验

哪个参数是 ASM 实例所必需的？

1. INSTANCE\_TYPE
2. ASM\_DISKGROUPS
3. LARGE\_POOL\_SIZE
4. 以上都不是

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 1

## 小测验

默认情况下，细粒度条带化用于 \_\_\_\_\_ 和 \_\_\_\_\_。

1. 数据文件
2. 控制文件
3. 临时文件
4. 联机重做日志
5. SPFILE

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2 和 4

## 小结

在本课中，您应该已经学会：

- 使用 SQL\*plus、asmcmd 和 Oracle Enterprise Manager 管理 ASM 实例
- 创建和删除 ASM 磁盘组
- 指定 ASM 兼容性属性
- 扩展 ASM 磁盘组
- 比较检索 ASM 元数据的方法

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 5 概览：管理 ASM 实例

本练习包含以下主题：

- 使用 `asmcmd` 创建 ASM 磁盘组
- 使用 EM 删除 ASM 磁盘组
- 查看 ASM 元数据

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 配置 Oracle Network 环境

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 使用 Enterprise Manager 执行以下操作：
  - 创建其它监听程序
  - 创建 Oracle Net 服务别名
  - 配置连接时故障转移
  - 控制 Oracle Net 监听程序
- 使用 tnsping 测试 Oracle Net 的连接
- 确定何时使用共享服务器以及何时使用专用服务器

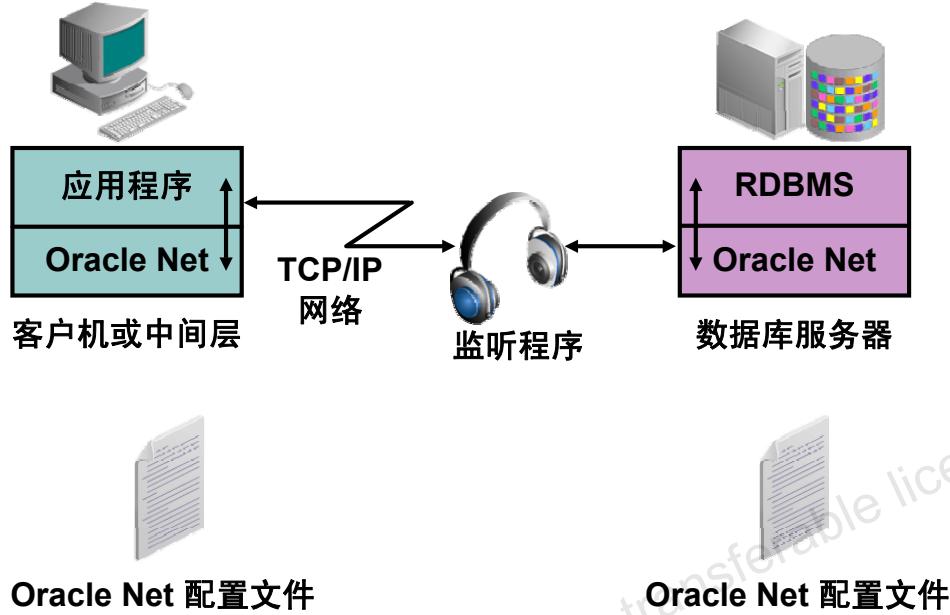
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 资源

- 《Oracle Database Net Services Administrator's Guide》
- 《Oracle Database Net Services Reference》

## Oracle Net 服务



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle Net 服务

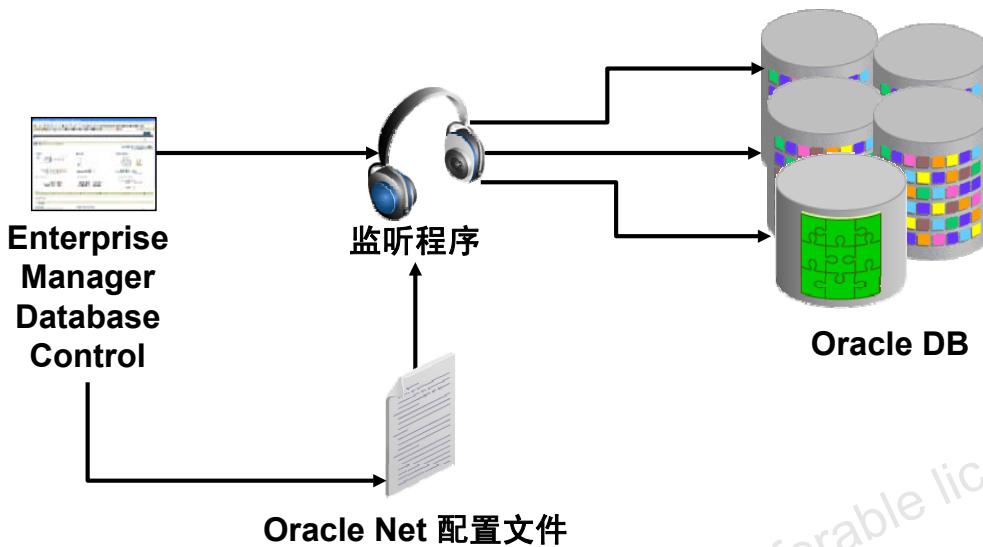
使用 Oracle Net 服务可建立从客户机或中间层应用程序到 Oracle 服务器的网络连接。建立网络会话之后，Oracle Net 充当客户机应用程序与数据库服务器的数据信使。它负责建立并维护客户机应用程序和数据库服务器之间的连接，并负责在二者之间交换消息。Oracle Net（或模拟 Oracle Net 功能的程序，如 Java 数据库连接）位于需要与数据库服务器通信的每台计算机上。

在客户机计算机上，Oracle Net 是供应用程序连接数据库的后台组件。

在数据库服务器上，Oracle Net 包含一个称为“Oracle Net 监听程序”的活动进程，该进程负责协调数据库与外部应用程序之间的连接。

Oracle Net 服务最常见的用法是传入数据库连接。通过配置其它网络服务，可允许访问外部代码库 (EXTPROC) 以及通过 Oracle 异构服务将 Oracle 实例连接到非 Oracle 数据源，如 Sybase、Informix、DB2 和 SQL Server。

## Oracle Net 监听程序



```
<Grid_home>/network/admin/listener.ora  
./sqlnet.ora
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle Net 监听程序

Oracle Net 监听程序（简称监听程序）是将所有非本地用户连接到 Oracle 实例的网关。单个监听程序可用于多个数据库实例以及成千上万个客户机连接。

Enterprise Manager 是访问监听程序的一种方式。可以控制实际监听程序以及口令保护和日志文件位置等一般参数的配置。

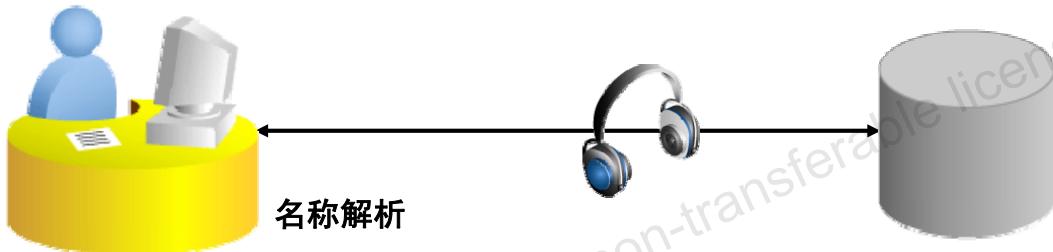
高级管理员还可以根据需要，使用标准操作系统 (OS) 文本编辑器（如 vi 或 gedit）手动编辑配置文件来配置 Oracle Net 服务。

**注：**安装了用于独立服务器的 Grid Infrastructure 后，Oracle Net 监听程序从其软件安装目录 (<Grid\_home>) 启动。需要从这一软件安装中运行监听程序，以便能够连接到 ASM 实例。默认使用这一监听程序监听安装在同一服务器上的所有数据库实例。

## 建立网络连接

要建立客户机或中间层连接，Oracle Net 要求客户机了解下列事项：

- 运行监听程序的主机
- 监听程序监视的端口
- 监听程序使用的协议
- 监听程序处理的服务名



版权所有 © 2010, Oracle。保留所有权利。

ORACLE

### 建立网络连接

如果要让应用程序通过 Oracle Net 监听程序连接到某服务，应用程序必须拥有关于该服务的信息，包括监听程序所驻留的地址或主机、监听程序接受的协议，以及监听程序监视的端口。在确定监听程序的位置之后，应用程序所需的最后一项信息就是它所要连接到的服务名。

Oracle Net “名称解析”就是确定该连接信息的过程。

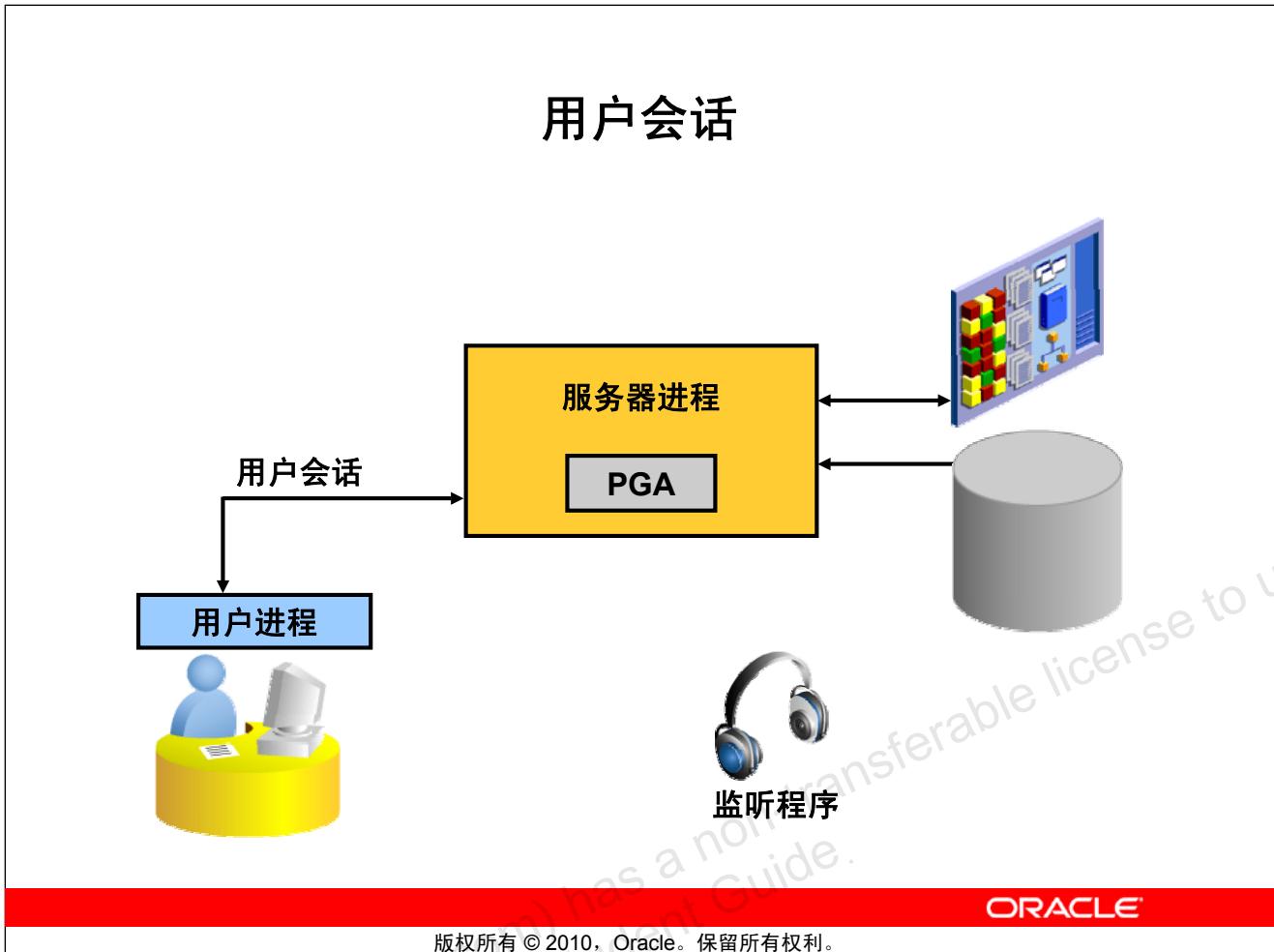
## 建立连接



### 建立连接

Oracle Net 名称解析完成之后，连接请求将从用户或中间层应用程序（以下称为“用户进程”）传递到监听程序。监听程序会接收一个 CONNECT 数据包，之后会检查此 CONNECT 数据包请求的 Oracle Net 服务名是否有效。

如果没有请求服务名（例如 tnsping 请求），监听程序会确认连接请求，此外不执行任何其它操作。如果请求的服务名无效，则监听程序会将错误代码传输给该用户进程。



### 用户会话

如果 CONNECT 数据包请求了一个有效的服务名，则监听程序将衍生一个新的进程来处理该连接。此新进程称为“服务器进程”。监听程序会连接到该进程并传递初始化信息，包括用户进程的地址信息。此时，监听程序不再处理连接，所有工作都会传递到服务器进程。

服务器进程将检查用户的验证身份证明（通常为口令），如果身份证明有效，则创建一个用户会话。

**专用服务器进程：**建立会话后，服务器进程随即充当服务器上的用户代理。服务器进程负责以下工作：

- 对通过应用程序发出的所有 SQL 语句进行语法分析，然后运行
- 在数据库缓冲区高速缓存中查找执行 SQL 语句所需的数据块
- 从磁盘上的数据文件中将必要的数据块读入到系统全局区 (SGA) 的数据库缓冲区高速缓存部分（如果 SGA 中没有这些数据块）
- 管理所有排序活动。排序区是用于处理排序的内存区；它包含在与程序全局区 (PGA) 关联的内存部分中
- 将结果返回到用户进程，以便应用程序可以处理这些信息
- 读取审计选项并将用户进程报告给审计目标

## 配置和管理 Oracle Network 的工具

- Enterprise Manager 的“Net Services Administration（网络服务管理）”页
- Oracle Net Manager
- Oracle Net Configuration Assistant
- 命令行



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 配置和管理 Oracle Network 的工具

使用以下工具和应用程序管理 Oracle Network 配置：

- **Enterprise Manager:** 提供了配置和管理 Oracle Net 服务的集成环境。使用 Enterprise Manager 可针对多个文件系统上的 Oracle 主目录配置 Oracle Net 服务，还可管理监听程序。
- **Oracle Net Manager:** 提供一个图形用户界面 (GUI)，通过此界面可针对本地客户机或服务器主机上的 Oracle 主目录配置 Oracle Net 服务。通过 Oracle Net Manager，可针对本地客户机或服务器主机上的 Oracle 主目录配置 Oracle Net 服务。可以使用 Oracle Net Manager 配置以下网络组件：
  - **命名:** 定义简单的名称和连接标识符，并将它们映射至连接描述符以确定网络位置和服务标识。Oracle Net Manager 支持在本地 `tnsnames.ora` 文件或集中式目录服务中配置连接描述符
  - **命名方法:** 配置将连接标识符解析为连接描述符的不同方法
  - **概要文件:** 配置用于在客户机或服务器上启用和配置 Oracle Net 功能的首选项
  - **监听程序:** 创建和配置监听程序以接收客户机连接

## 配置和管理 Oracle Network 的工具（续）

- **Oracle Net Configuration Assistant:** 安装 Oracle 软件时通过 Oracle Universal Installer 启动。使用 Oracle Net Configuration Assistant 可配置 Oracle DB 的监听协议地址和服务信息。在典型的数据库安装期间，Oracle Net Configuration Assistant 会自动配置一个名为 LISTENER 的监听程序，该监听程序具有数据库的 TCP/IP 监听协议地址。如果执行定制安装，则 Oracle Net Configuration Assistant 会提示配置所选择的监听程序名称和协议地址。在安装数据库之后，使用 Oracle Net Configuration Assistant 进行初始网络配置。然后，可以使用 Oracle Enterprise Manager 和 Oracle Net Manager 配置和管理您的网络。
- **命令行:** 用于启动、停止监听程序进程或查看监听程序进程的状态。由操作系统用户（在本课程中为 oracle）启动或停止监听程序。如果未启动监听程序，则无法使用 Enterprise Manager。

## 监听程序控制实用程序

可以使用 lsnrctl 命令行实用程序（或通过 EM）控制 Oracle Net 监听程序。

```
$ . oraenv
ORACLE_SID = [orcl] ? +ASM
$ lsnrctl

LSNRCTL for Linux: Version 11.2.0.1.0 - Production on 30-JUN-2009 00:47:01
Copyright (c) 1991, 2009, Oracle. All rights reserved.
Welcome to LSNRCTL, type "help" for information.

LSNRCTL> help
The following operations are available
An asterisk (*) denotes a modifier or extended command:
start          stop          status
services       version       reload
save_config    trace         spawn
change_password quit         exit
set*           show*
```

版权所有 © 2010, Oracle。保留所有权利。

### 监听程序控制实用程序

启动实例时，监听程序进程会建立一个指向 Oracle DB 的通信路径。随后，监听程序可接受数据库连接请求。

使用监听程序控制实用程序可控制监听程序。使用 lsnrctl，可以：

- 启动监听程序
- 停止监听程序
- 检查监听程序的状态
- 根据配置文件参数重新初始化监听程序
- 动态配置多个监听程序
- 更改监听程序口令

该实用程序的基本命令语法为：

LSNRCTL> *command* [*listener\_name*]

发出 lsnrctl 命令时，如果没有指定其它监听程序名称或执行 SET CURRENT\_LISTENER 命令，此命令将作用于默认的监听程序（名为 LISTENER）。如果监听程序名为 LISTENER，则可省略 *listener\_name* 参数。幻灯片中显示了 lsnrctl 的有效命令。

**注：**在 Grid Infrastructure 主目录和 Oracle DB 主目录中都有 lsnrctl 实用程序。在使用此实用程序之前，务必将环境变量设置为合适的主目录。

## 监听程序控制实用程序的语法

可以使用命令行或 LSNRCTL 提示符发出监听程序控制实用程序的命令。

- 命令行语法:

```
$ lsnrctl <command name>
$ lsnrctl start
$ lsnrctl status
```

- 提示符语法:

```
LSNRCTL> <command name>
LSNRCTL> start
LSNRCTL> status
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 监听程序控制实用程序的语法

可以从实用程序内部（提示符语法）或命令行发出 lsnrctl 命令。以下两个命令具有相同的效果，但分别使用的是命令行语法和提示符语法：

**命令行语法:**

```
$ lsnrctl start
```

**提示符语法:**

```
$ lsnrctl
LSNRCTL for Linux: Version 11.2.0.1.0 - Production on 30-JUN-
2009 01:00:01
Copyright (c) 1991, 2009, Oracle. All rights reserved.
Welcome to LSNRCTL, type "help" for information.
LSNRCTL> start
```

命令行语法通常用于执行单个命令或脚本命令。如果计划执行若干个连续的 lsnrctl 命令，则使用提示符语法更高效。请注意，上面省略了 listener\_name 参数，因此 stop 命令会影响名为 LISTENER 的监听程序。如果监听程序受到口令的保护，则必须使用提示符语法。

## 监听程序控制实用程序的语法（续）

请注意，如果监听程序名称不是 LISTENER，则在命令语法中必须包含监听程序名称，或者使用 SET CURRENT\_LISTENER 命令。假设您的监听程序名为 custom\_lis。以下是使用提示符语法停止名为 custom\_lis 的监听程序的两个示例：

```
LSNRCTL> stop custom_lis
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=host01) (PORT=5521)))
The command completed successfully
```

该示例与以下示例产生的结果相同：

```
LSNRCTL> set cur custom_lis
Current Listener is custom_lis
LSNRCTL> stop
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=host01) (PORT=5521)))
The command completed successfully
```

注：在上面的语法中，current\_listener 缩写成了 cur。

使用命令行语法可产生相同的结果：

```
$ lsnrctl stop custom_lis
LSNRCTL for Linux: Version 11.2.0.1.0 - Production on 30-JUN-
2009 01:01:53
Copyright (c) 1991, 2009, Oracle. All rights reserved.
Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=host01) (PORT=5521)))
The command completed successfully
```

## 使用 SRVCTL 启动和停止监听程序

如果配置了 Oracle Restart 来监视您的监听程序，则您应使用 SRVCTL 来管理该监听程序。

- 语法示例：

```
$ srvctl -h  
$ srvctl start listener  
$ srvctl stop listener  
$ srvctl start listener -l mylistener  
$ srvctl status listener
```



版权所有 © 2010, Oracle。保留所有权利。

## 使用 SRVCTL 启动和停止监听程序

对于由 Oracle Restart 管理的任何监听程序，可以使用 SRVCTL 启动、停止它并查看其状态。示例包括：

- 使用 SRVCTL 中提供的命令显示帮助：`srvctl -h`
- 启动默认监听程序：`srvctl start listener`
- 停止默认监听程序：`srvctl stop listener`
- 启动名为 mylistener 的监听程序：`srvctl start listener -l mylistener`
- 显示默认监听程序的状态：`srvctl status listener`

## 监听程序主页

**General**

Status Up  
Up Since Jun 19, 2009 2:02:00 AM GMT+07:00  
Instance Name orcl  
Version 11.2.0.1.0  
Host edrsr25p1.us.oracle.com  
Listener LISTENER edrsr25p1.us.oracle.com  
ASM +ASM edrsr25p1.us.oracle.com

[View All Properties](#)

**General**

Status Up  
Availability (%) 100 (Last 24 Hours)  
Alias LISTENER  
Version 11.2.0.1.0  
Oracle Home /u01/app/oracle/product/11.2.0/grid  
Net Address (ADDRESS=(PROTOCOL=TCP)(HOST=edrsr25p1.us.oracle.com)(PORT=1521))  
LISTENER.ORA Location /u01/app/oracle/product/11.2.0/grid/network  
Start Time Jun 18, 2009 3:20:31 AM  
Host edrsr25p1.us.oracle.com  
Oracle Restart Enabled

**State**

TNS Ping (ms) ✓ 10  
Established Connections per minute 2.2  
Refused Connections per minute 0

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## 监听程序主页

单击 Enterprise Manager 的“Database（数据库）”主页上的“Listener（监听程序）”链接可访问“Listener（监听程序）”主页。

在此页上，可看到：

- 最近 24 小时内监听程序的状态和可用性
- 监听程序版本和 Oracle 主目录
- 监听程序的第一个监听地址
- 用于启动监听程序的配置文件的位置
- 监听程序的启动时间和主机信息
- Oracle Restart 状态

要启动监听程序，请转到“Database（数据库）”主页，单击监听程序名称以打开“Listener（监听程序）”主页。单击“Stop（停止）”可停止运行的监听程序；单击“Start（启动）”可启动未运行的监听程序。请以可启动和停止监听程序的操作系统用户身份登录到主机。

## “Net Services Administration (网络服务管理)”页

ORAQUE Enterprise Manager 11g Database Control Host edrsr25p1.us.oracle.com > Net Services Administration

Net Services Administration allows you to configure or administer the following network components:

- Listener: Allows configuration and administration functions on listeners.
- Directory Naming: Allows configuration and administration of Net service names on a Directory server.
- Local Naming: Allows configuration and administration of Net service names on a client's tnsnames.ora file.
- Network Profile: Allows configuration of preferences for Oracle Net Services features on the client or server.
- File Location: Allows specification of the configuration file location for the Oracle Home.

Choose a configuration file location, then select the feature that you want to administer and click 'Go'.

Administer	Listeners	Go
Select	Con	tion
<input checked="" type="radio"/> /u01	Directory Naming	Oracle Home
	Local Naming	/u01/app/oracle/product/11.2.0/db_home1
	Network Profile	
	File Location	

Database | Setup | Preferences | Help | Logout

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### “Net Services Administration (网络服务管理)”页

使用“Net Services Administration (网络服务管理)”页可针对多个文件系统上的 Oracle 主目录配置 Oracle Net 服务。此页还提供管理监听程序的公用管理功能。可以使用“Net Services Administration (网络服务管理)”页配置和管理以下各项：

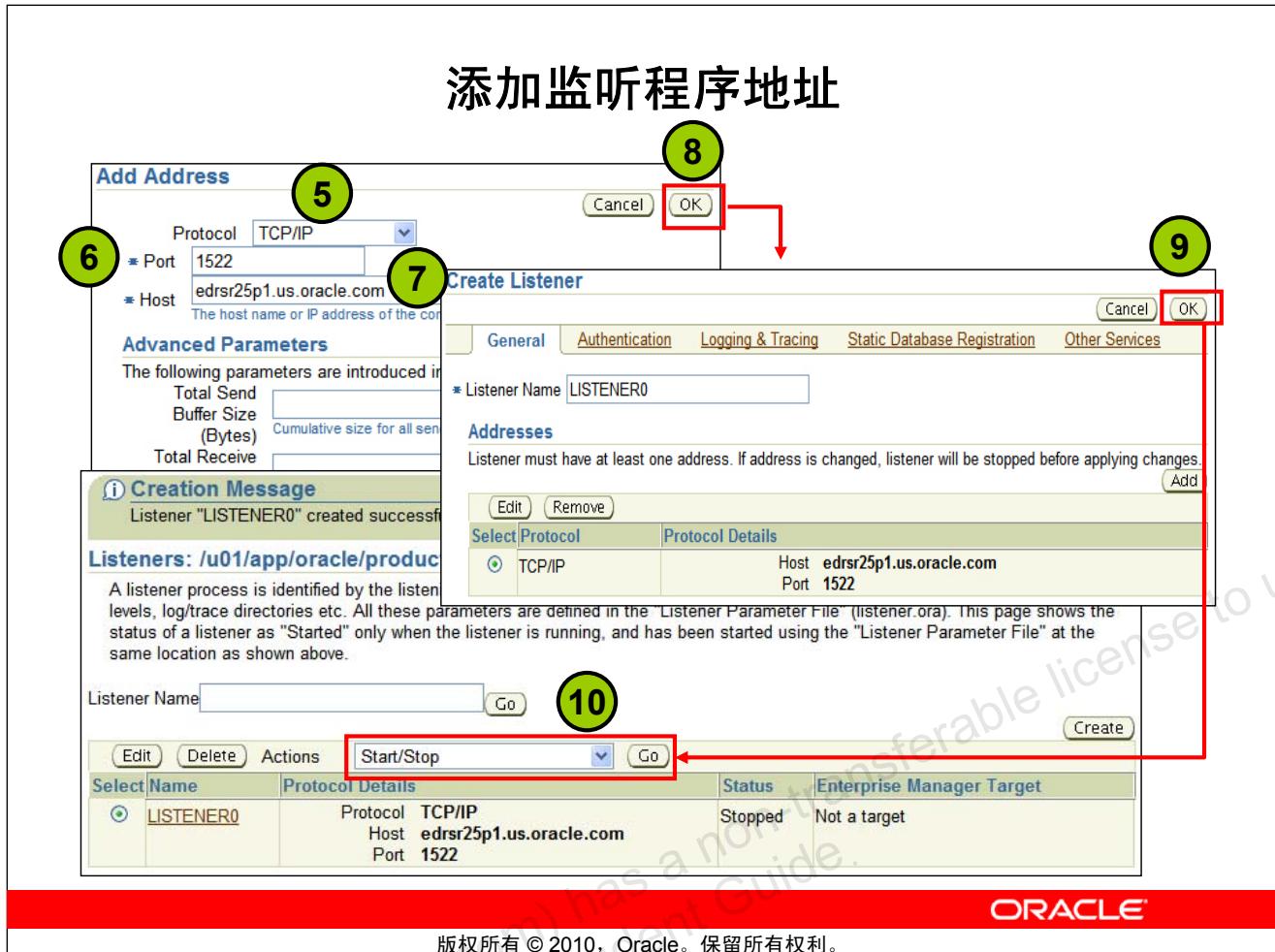
- **监听程序：**添加、删除、启动和停止监听程序，以及更改其跟踪特性和事件记录特性。此外，还可查看监听程序的控制状态报表。
- **目录命名：**可以定义简单的名称和连接标识符，并将定义的内容映射至连接描述符以确定网络位置和服务标识。将数据库服务、Oracle Net 服务和 Oracle Net 服务别名保存到集中式目录服务中。
- **本地命名：**将 Oracle Net 服务名保存在 tnsnames.ora 文件中。
- **概要文件：**配置 sqlnet.ora 参数。
- **文件位置：**更改 Oracle Net 服务的配置文件的位置。



## 创建监听程序

要创建 Oracle Net 监听程序，请单击“Listener（监听程序）”属性页“Related Links（相关链接）”区域中的“Net Services Administration（网络服务管理）”。然后执行以下步骤：

1. 从“Administer（管理）”下拉列表中选择“Listeners（监听程序）”，然后单击“Go（执行）”。
2. 如果未输入主机首选身份证明，则出现“Host Login（主机登录）”页。输入用户名和口令，然后单击“Login（登录）”。
3. 单击“Create（创建）”。
4. 输入一个监听程序名称。此名称对此服务器必须是唯一的。单击“Add（添加）”添加一个监听程序地址。每个监听程序必须至少有一个监听程序地址。

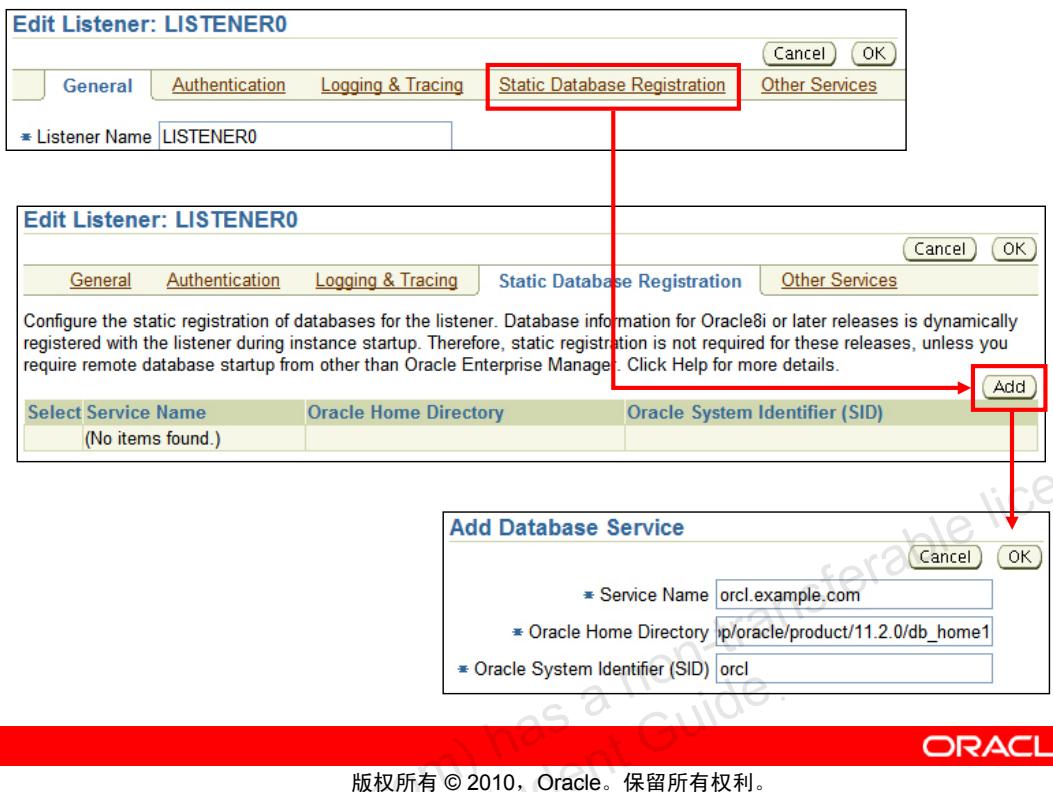


## 添加监听程序地址

创建监听程序工作流的下一步是创建监听程序地址：

5. 选择网络协议。TCP/IP 是最常用的协议，也是默认协议。可以选择其它协议，如 Internal Process Communication（内部进程通信，IPC），此协议通常用于连接到本地应用程序（驻留于数据库服务器上），或者选择外部代码库 (EXTPROC) 以及带有 SSL 的 TCP/IP。
- 注：使用“Other Services（其他服务）”选项卡可配置 EXTPROC 协议。
6. 输入监听程序要监视的端口。Oracle Net 的默认端口是 1521。如果选择使用 1521 之外的端口，该监听程序或实例还需要进行其它配置。
7. 输入监听程序将在其上运行的服务器的名称或 IP 地址。
8. 对监听程序而言，其它所有配置步骤都是可选项。单击“OK（确定）”保存地址。唯一必不可少的配置是监听地址和名称。
9. 在“Create Listener（创建监听程序）”页上，复查刚创建的地址的有关信息，然后单击“OK（确定）”保存更改。
10. 要启动新的监听程序，请从“Actions（操作）”下拉列表中选择“Start/Stop（启动/停止）”，然后单击“Go（执行）”。

## 数据库服务注册



### 数据库服务注册

监听程序必须了解实例名称以及实例的 ORACLE\_HOME 位置，然后才能将客户机连接转发到实例。监听程序可通过下列两种方式查找此信息：

- 动态服务注册：** Oracle8i 和更高版本的实例会在启动数据库时向默认监听程序自动进行注册。默认监听程序不需要其它的监听程序配置。
- 静态服务注册：** Oracle DB 的早期版本并不自动向监听程序进行注册，因此，要求监听程序配置文件包含一个列表，其中含有监听程序为之提供服务的所有数据库服务。在较新版本中，如果存在以下情况，仍然可以选择使用静态服务注册：
  - 监听程序所在的端口不是默认的 1521 端口，您又不希望将实例配置为使用非默认端口进行注册
  - 应用程序要求进行静态服务注册

要添加静态数据库服务，请单击“Edit Listener（编辑监听程序）”页上的“Static Database Registration（静态数据库注册）”，然后单击“Add（添加）”按钮。输入服务名称（与全局数据库名 <DB\_NAME>.<DB\_DOMAIN> 相同）、ORACLE\_HOME 路径和 SID（与实例名相同）。单击“OK（确定）”。要使这些更改生效，必须重新加载（使用 RELOAD 命令）或重新启动监听程序。

## 数据库服务注册（续）

### 服务名

SERVICE\_NAMES 初始化参数指定客户机可用来连接到实例的一个或多个名称。实例向监听程序注册其服务名。客户机请求服务时，监听程序确定哪些实例提供所请求的服务，并将客户机路由到相应的实例。

您可以指定多个服务名以区分同一数据库的不同用法，如下面的示例所示：

```
SERVICE_NAMES = sales.example.com, eurosales.example.com
```

还可以使用服务名来标识两个不同的数据库通过复制提供的单个服务。

如果未在该参数中使用域来限定名称，则 Oracle 会使用 DB\_DOMAIN 参数的值来限定它们。如果未指定 DB\_DOMAIN，则不会向非限定的 SERVICE\_NAMES 值应用任何域。

处理客户机连接请求时，监听程序尝试将该参数的值与客户机连接描述符中 SERVICE\_NAME 参数的值进行匹配。如果客户机连接描述符使用 SID 参数，则监听程序不会尝试映射这些值。该参数的值通常来自初始化参数文件中 DB\_NAME 和 DB\_DOMAIN 参数的组合 (DB\_NAME.DB\_DOMAIN)，但此值也可以包含客户机用于标识服务的任何有效名称。

## 命名方法

Oracle Net 支持多种解析连接信息的方法：

- 简便连接命名：使用 TCP/IP 连接字符串
- 本地命名：使用本地配置文件
- 目录命名：使用符合 LDAP 的集中式目录服务器
- 外部命名：使用受支持的非 Oracle 命名服务



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 命名方法

Oracle Net 支持以下几种命名方法：

- **简便连接命名：** 使用简便连接命名方法，客户机可使用如下所示的 TCP/IP 连接字符串（由主机名、可选端口和服务名组成）连接到 Oracle DB 服务器：

```
CONNECT username/password@host[:port] [/service_name]
```

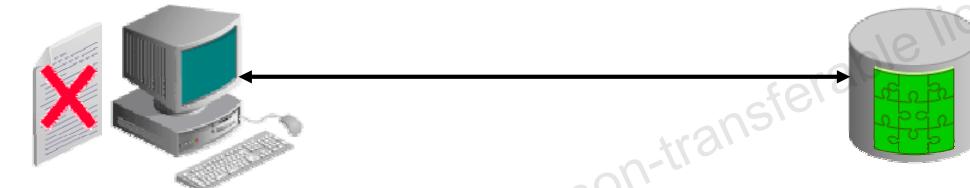
简便连接命名方法不需要进行配置。

- **本地命名：** 本地命名方法将连接描述符（由网络服务名标识）存储在客户机上的名为 `tnsnames.ora` 的本地配置文件中。
- **目录命名：** 为了访问数据库服务，目录命名方法将连接标识符存储在符合轻型目录访问协议 (LDAP) 的集中式目录服务器中。
- **外部命名：** 外部命名方法将网络服务名存储在受支持的非 Oracle 命名服务中。受支持的第三方服务包括：
  - 网络信息服务 (NIS) 外部命名
  - 分布式计算环境 (DCE) 单元目录服务 (CDS)

## 简便连接

- 默认启用此方式
- 不需要进行客户机配置
- 仅支持 TCP/IP (无 SSL)
- 不支持高级连接选项, 如:
  - 连接时故障转移
  - 源路由
  - 负载平衡

```
SQL> CONNECT hr/hr@db.us.oracle.com:1521/dba11g
```



无 Oracle Net 配置文件

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 简便连接

使用简便连接时, 您需要在连接字符串中提供 Oracle Net 连接所需的所有信息。简便连接的连接字符串采用以下形式:

<username>/<password>@<hostname>:<listener port>/<service name>

监听程序端口和服务名为可选项。如果未提供监听程序端口, Oracle Net 假定使用的是默认端口 1521。如果未提供服务名, Oracle Net 假定连接字符串中提供的数据库服务名与主机名是相同的。

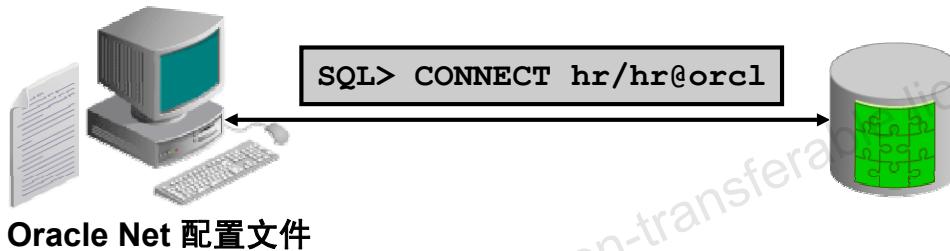
假定监听程序使用 TCP 监听端口 1521, 并使用 SERVICE\_NAMES=db 和 DB\_DOMAIN=us.oracle.com 实例参数, 则幻灯片中显示的连接字符串可简化为:

SQL> connect hr/hr@db.us.oracle.com

**注:** SERVICE\_NAMES 初始化参数可接受多个逗号分隔的值。这些值中只能有一个 db, 此方案才能起作用。

## 本地命名

- 需要客户机名称解析文件
- 支持所有的 Oracle Net 协议
- 支持高级连接选项，如：
  - 连接时故障转移
  - 源路由
  - 负载平衡



版权所有 © 2010, Oracle。保留所有权利。

ORACLE

### 本地命名

使用本地命名时，用户可为 Oracle Net 服务提供别名。Oracle Net 会根据本地已知服务的列表来检查别名，如果发现匹配名称，会将别名转换为主机、协议、端口和服务名。

本地命名的一个优势是，数据库用户仅需要记住一个短别名，而不必记住简便连接所需的长连接字符串。

本地已知服务列表存储在以下文本配置文件中：

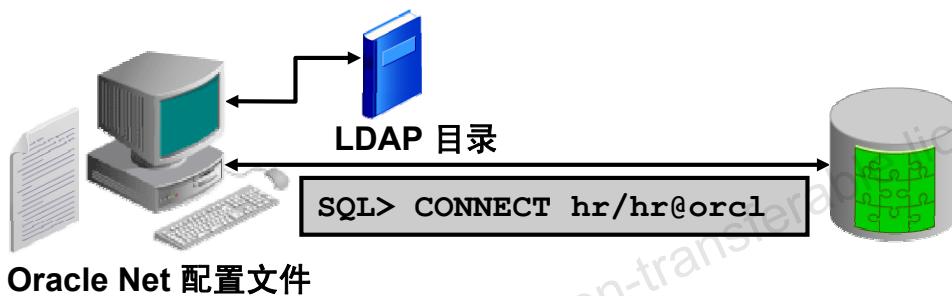
<oracle\_home>/network/admin/tnsnames.ora

这是 tnsnames.ora 文件的默认位置，但是，也可以使用 TNS\_ADMIN 环境变量将该文件放到其它位置。

如果组织的 Oracle Net 服务配置不经常更改，则适合使用本地命名。

## 目录命名

- 需要加载了 Oracle Net 名称解析信息的 LDAP:
  - Oracle Internet Directory
  - Microsoft Active Directory Services
- 支持所有的 Oracle Net 协议
- 支持高级连接选项



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 目录命名

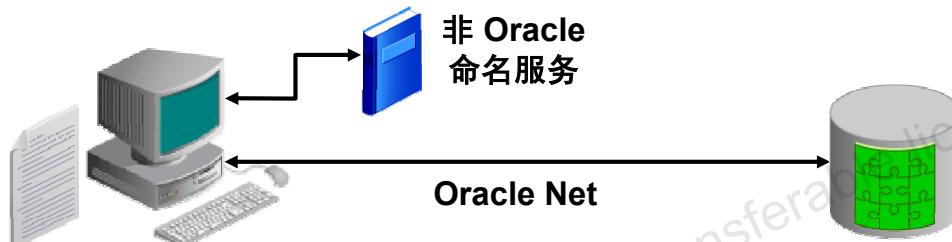
使用目录命名时，用户可为 Oracle Net 服务提供别名。Oracle Net 会根据外部已知服务的列表来检查别名，如果发现匹配名称，会将别名转换为主机、协议、端口和服务名。与本地命名一样，数据库用户只需要记住短别名。

目录命名的一个优势是，一旦将新的服务名添加到 LDAP 目录，此服务名便可供用户在连接时使用。使用本地命名时，必须先由数据库管理员 (DBA) 分发更新的 `tnsnames.ora` 文件（这些文件包含已更改的服务名信息），用户才能连接到新服务或修改后的服务。

如果组织的 Oracle Net 服务配置经常更改，则适合使用目录命名。

## 外部命名方法

- 使用受支持的非 Oracle 命名服务
- 包括:
  - 网络信息服务 (NIS) 外部命名
  - 分布式计算环境 (DCE) 单元目录服务 (CDS)



ORACLE

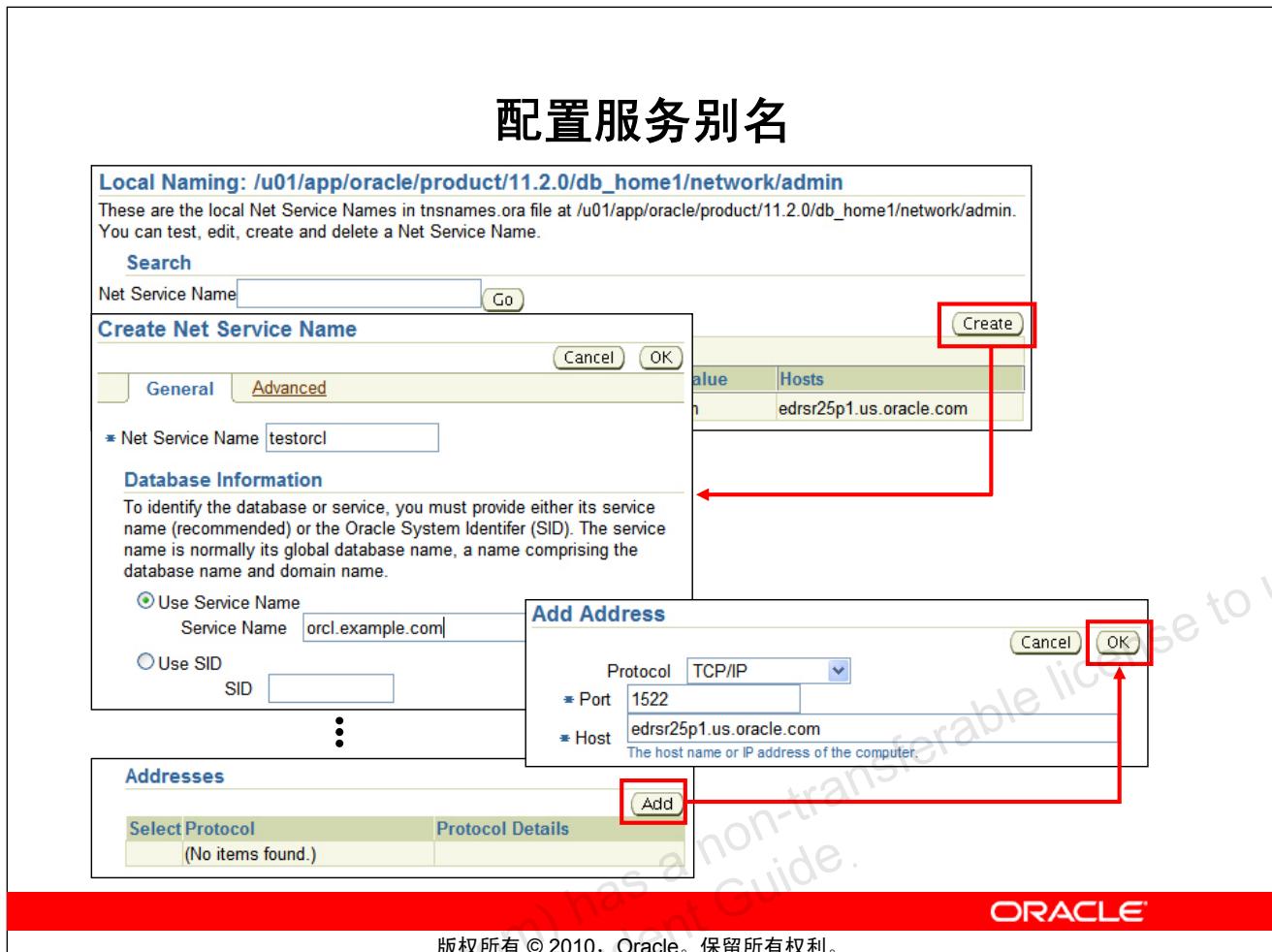
版权所有 © 2010, Oracle。保留所有权利。

### 外部命名方法

外部命名方法将 Oracle Net 服务名存储在受支持的非 Oracle 命名服务中。受支持的第三方服务包括：

- 网络信息服务 (NIS) 外部命名
- 分布式计算环境 (DCE) 单元目录服务 (CDS)

从概念上讲，外部命名类似于目录命名。



## 配置服务别名

要创建本地 Oracle Net 服务别名，请从“Net Services Administration（网络服务管理）”页上的“Administer（管理）”下拉列表中选择“Local Naming（本地命名）”，然后单击“Go（执行）”。然后单击“Create（创建）”。

通过选择“Directory Naming（目录命名）”而不是“Local Naming（本地命名）”，可配置目录命名的服务别名。

**注：**如果尚未配置目录命名，则无法选择“Directory Naming（目录命名）”选项。在“Oracle 企业身份管理”课程以及《Oracle Advanced Security Administration》手册中对目录命名进行了介绍。

在“Create Net Service Name（创建网络服务名）”页的“Net Service Name（网络服务名）”字段中输入一个唯一名称。（此名称是用户输入的要使用的别名）。输入要连接到的数据库的服务名或系统标识符(SID)，然后单击“Add（添加）”按钮，输入服务名的地址。

对于地址，输入监听程序连接所需服务要使用的协议、端口和主机。

## 高级连接选项

对于本地命名和目录命名，Oracle Net 支持下列高级连接选项：

- 连接时故障转移
- 源路由
- 负载平衡

Select Protocol	Protocol Details
<input checked="" type="radio"/> TCP/IP	Host edrsr25p1.us.oracle.com Port 1522
<input type="radio"/> TCP/IP	Host edrsr25p1.us.oracle.com Port 1521

**Connect-time Failover and Client Load Balancing**

Configure whether addresses are tried randomly or sequentially during connections to the service. This setting is applicable only if there are more than one addresses configured.

Try each address, in order, until one succeeds  
 Try each address randomly, until one succeeds  
 Try one address, selected at random  
 Use each address in order until destination is reached  
 Use only the first address

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 高级连接选项

可以通过多个监听程序协议地址访问某个数据库服务时，可以指定使用这些地址的顺序。可以随机选择这些地址，也可以依次尝试这些地址。如果有多个监听程序可用（如 Oracle Real Application Cluster (RAC) 配置），Oracle Net 可以利用监听程序故障转移和负载平衡功能以及 Oracle Connection Manager 源路由功能。

如果启用了“连接时故障转移”，则一个别名下将列出两个或多个监听程序地址。如果第一个地址不可用，则尝试使用第二个地址。Oracle Net 会始终按列出的顺序试用地址，直至遇到起作用的监听程序，或者试用的所有地址均失败。透明应用程序故障转移 (TAF) 是一项客户机功能，它使客户机可以在数据库实例发生故障时重新连接到正常运行的数据库。服务器使用通知在客户机上触发 TAF 回调。

如果启用了“负载平衡”，Oracle Net 会从地址列表中随机选择一个地址。运行时连接负载平衡功能可以在多个分派程序之间平衡活动的连接数，从而改进连接性能。在 RAC 环境中，连接池负载平衡还能在多个实例之间平衡活动的连接数。

## 高级连接选项（续）

“源路由”与 Oracle Connection Manager 一起使用，后者充当 Oracle Net 通信的代理服务器，使 Oracle Net 通信可以安全地通过防火墙进行路由。Oracle Net 将地址视为中继列表，首先连接到第一个地址，然后请求从第一个地址传递到第二个地址，直至到达目标位置。它与故障转移或负载平衡的不同之处在于，每次进行连接时都会用到所有的地址。

选项	高级功能
按顺序尝试每个地址，直至成功为止。	故障转移
随机尝试每个地址，直至成功为止。	故障转移 负载平衡
随机尝试所选的一个地址。	负载平衡
按顺序使用每个地址，直至到达目标位置。	源路由
仅使用第一个地址。	无

## 测试 Oracle Net 连接

tnsping 实用程序测试 Oracle Net 服务别名：

- 确保客户机与 Oracle Net 监听程序之间的连接
- 不验证所请求的服务是否可用
- 支持简便连接名称解析：

```
tnsping host01.example.com:1521/orcl
```

- 支持本地命名和目录命名：

```
tnsping orcl
```



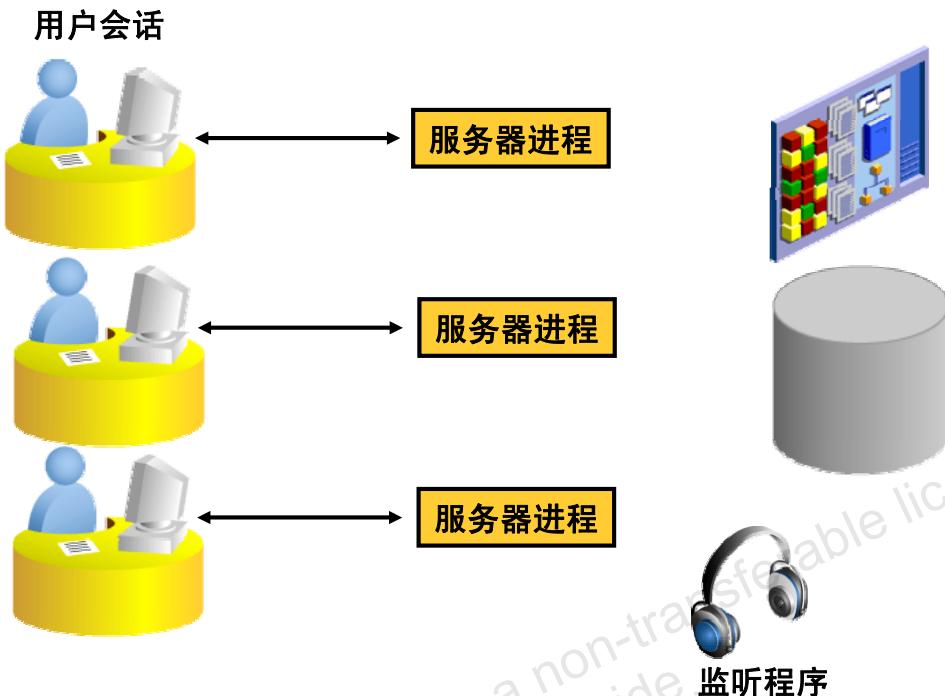
版权所有 © 2010, Oracle。保留所有权利。

### 测试 Oracle Net 连接

Oracle Net 中的 tnsping 相当于 TCP/IP 的 ping 实用程序。它提供了快速测试，可验证到目标位置的网络路径是否正常。例如，在命令行窗口中输入 tnsping orcl。

该实用程序可验证主机名、端口以及协议是否可到达监听程序，它并不实际检查监听程序是否处理服务名。tnsping 实用程序还会显示配置文件的位置。当系统中具有多个 ORACLE\_HOME 位置时，这很有用。

## 用户会话：专用服务器进程



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

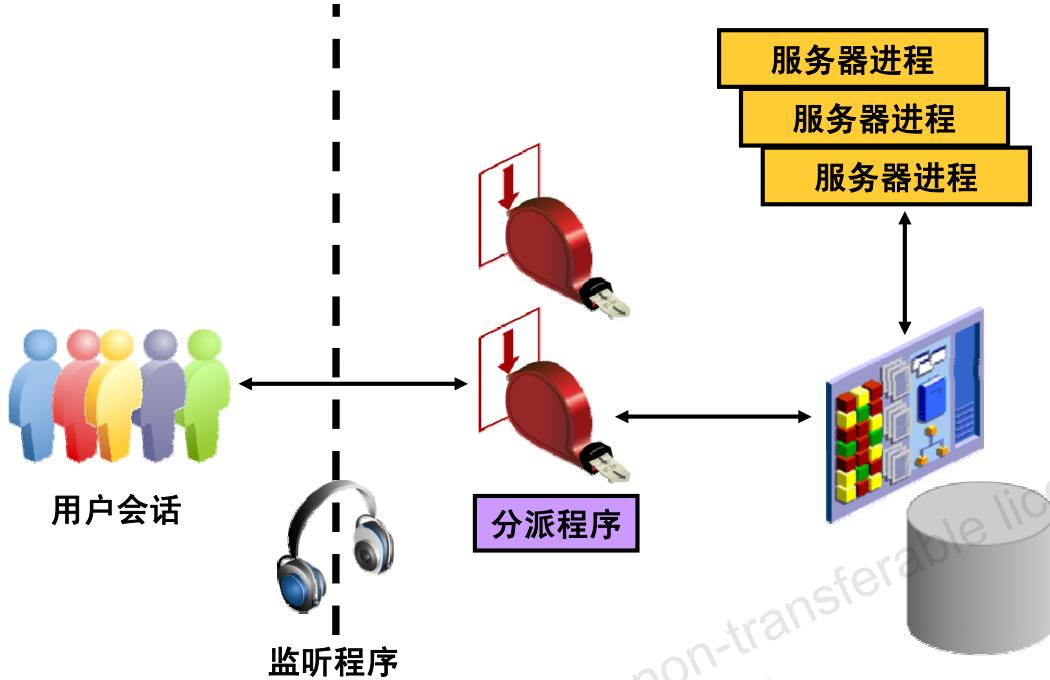
### 用户会话：专用服务器进程

使用专用服务器进程时，服务器进程与用户进程之间的比例是一比一。每个服务器进程都会使用系统资源，包括 CPU 周期和内存。

在负荷很高的系统中，由于专用服务器进程占用了内存和 CPU 资源，其成本可能难以承受，还可能对系统的可伸缩性产生负面影响。如果专用服务器体系结构的资源需求对系统产生了负面影响，则您有下列两种选择：

- 通过增加更多的内存和额外的 CPU 容量来增加系统资源
- 使用 Oracle 共享服务器进程体系结构

## 用户会话：共享服务器进程



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 用户会话：共享服务器进程

每个参与共享服务器进程体系结构的服务均至少具有一个（通常为多个）分派程序进程。连接请求到达时，监听程序将不衍生专用服务器进程，而是维护一个可用于每个服务名的分派程序以及每个分派程序的连接负载（并发连接数量）列表。

连接请求将路由至负载最小的分派程序，该程序将为给定服务名提供服务。用户在会话持续时间内均一直连接至相同的分派程序。

与专用服务器进程不同，单个分派程序可以管理数以百计的用户会话。

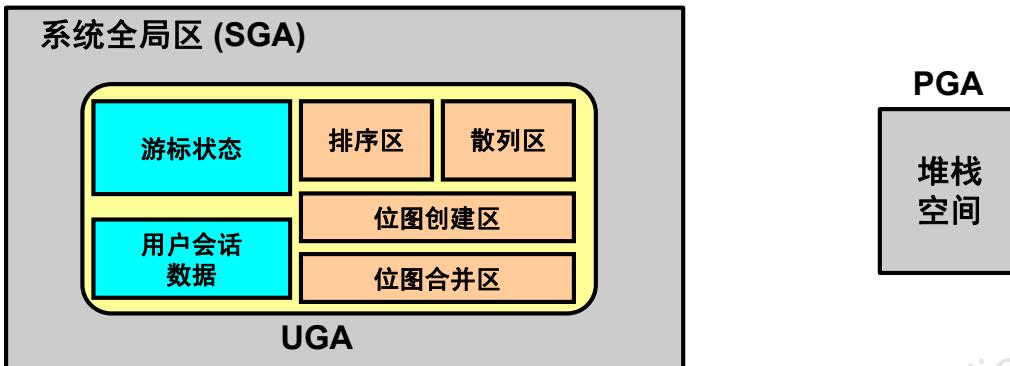
实际上分派程序并不处理用户请求的工作，而是将用户请求传递到位于 SGA 共享池部分的公用队列。

共享服务器进程接管专用服务器进程的大部分工作，将请求从队列中拉出并进行处理，直至完成该请求。

由于单个用户会话的请求可能由多个共享服务器进程来处理，因此通常存储在 PGA 中的大部分内存结构必须位于共享内存位置（默认情况下在共享池中）。但是，如果配置了大型池，或者为自动内存管理设置了 SGA\_TARGET，则这些内存结构会存储在 SGA 的大型池部分。

## SGA 和 PGA

Oracle 共享服务器：在 SGA 中存放用户会话数据。



调整 SGA 大小时，请务必考虑共享服务器内存要求。

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### SGA 和 PGA

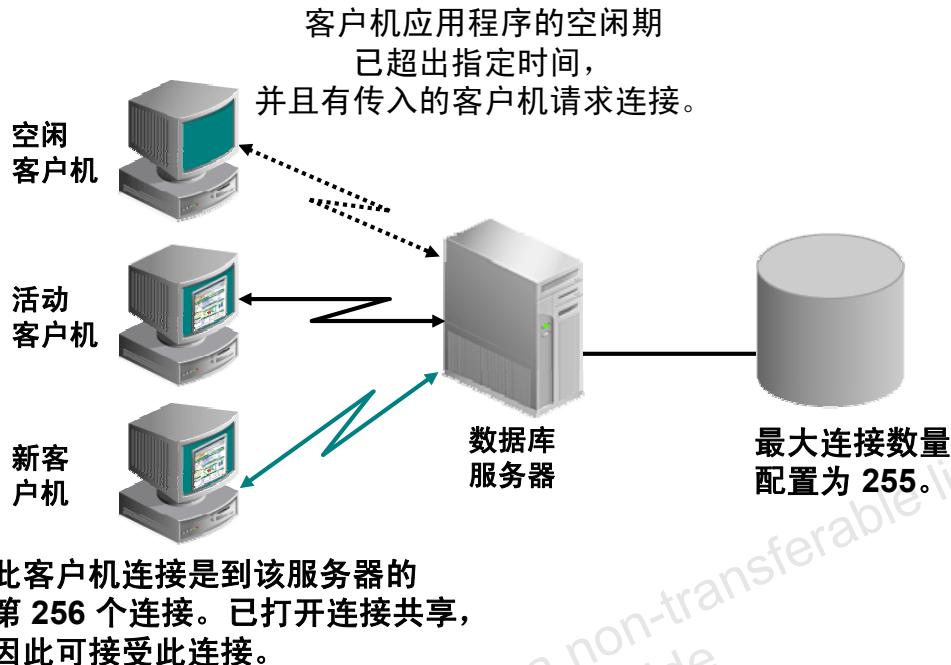
使用专用服务器或共享服务器时，SGA 和 PGA 的内容不同：

- 所有 SQL 语句的文本和语法分析形式均存储在 SGA 中。
- 游标状态包含 SQL 语句的运行时内存值，例如检索的行等。
- 用户会话数据包括安全性和资源用量信息。
- 堆栈空间包含进程的局部变量。

### 技术注释

SGA 和 PGA 中的更改对用户而言是透明的；但是，如果要支持多个用户，则需要增大 `LARGE_POOL_SIZE` 初始化参数。每个共享服务器进程均必须访问所有会话的数据空间，以便任何服务器均可以处理来自任何会话的请求。在 SGA 中为每个会话分配数据空间。可通过设置 `PRIVATE_SGA` 资源限制参数来限制每个会话可以分配的空间数量，该参数位于用户概要文件“General（一般信息）”页的“Database Services（数据库服务）”区域中。

## 共享服务器：连接共享



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 共享服务器：连接共享

使用连接共享功能，数据库服务器可以使空闲会话超时，从而使用其连接为活动会话提供服务。空闲会话逻辑上仍处于打开状态，当该会话下次发出请求时会自动重新建立物理连接。因此，Web 应用程序可使用现有硬件来满足大量并发用户的需求。连接共享是通过共享服务器进行配置的。

在本示例中，Oracle DB 服务器配置了 255 个连接。某个客户机的空闲期已超出了指定时间。通过连接共享，此连接可用于第 256 个传入的客户机连接。如果空闲客户机还有其它任务要执行，通过另一个客户机的空闲连接可为该客户机重新建立连接。

## 不能使用共享服务器的情况

不能使用共享服务器执行某些类型的数据库工作：

- 数据库管理
- 备份和恢复操作
- 批处理和批量加载操作
- 数据仓库操作



分派程序



专用服务器进程

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 不能使用共享服务器的情况

Oracle 共享服务器体系结构是一个有效的进程和内存使用模型，但是并不适用于所有的连接。由于存在公共请求队列，并且大量用户会共享分派程序响应队列，所以当存在必须处理大批数据的操作时共享服务器的性能并不是很好，如处理仓库查询或执行批处理时。

使用 Oracle Recovery Manager 进行备份和恢复会话时（将在以后的课程中介绍），也会处理大量的数据集，因此也必须使用专用连接。

许多管理任务不能（无法）使用共享服务器连接执行，其中包括启动和关闭实例、创建表空间和数据文件、维护索引和表、分析统计信息，以及通常由 DBA 执行的其它多个任务。所有 DBA 会话都必须选择专用服务器。

## 配置数据库之间的通信

- 在站点之间发送数据或消息时需要在双方站点上进行网络配置。
- 您必须配置以下项：
  - 网络连接（例如 TNSNAMES.ora）
  - 数据库链接

```
CREATE DATABASE LINK <remote_global_name>
CONNECT TO <user> IDENTIFIED BY <pwd>
USING '<connect_string_for_remote_db>';
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 配置数据库之间的通信

数据库链接是一个数据库中的方案对象，通过该对象可以访问另一个数据库中的对象。另一个数据库不必是 Oracle DB 系统。但是，要访问非 Oracle 系统，必须使用 Oracle 异构服务。

要创建专用数据库链接，必须具有 CREATE DATABASE LINK 系统权限。要创建公用数据库链接，必须具有 CREATE PUBLIC DATABASE LINK 系统权限。还必须具有对远程 Oracle DB 的 CREATE SESSION 系统权限。

应用程序使用数据库链接访问远程数据库时，Oracle DB 会代表本地请求在远程数据库中建立一个数据库会话。创建数据库链接时使用的 CONNECT TO 子句确定了在远程数据库上建立连接的方式。可以创建固定用户、当前用户和已连接用户的数据库链接。当前用户链接只能通过 Oracle 高级安全选件使用。此幻灯片中的示例说明了用于创建固定用户数据库链接的语法。

创建数据库链接之后，可以使用它来引用另一个数据库上的表和视图。在 SQL 语句中，可以通过将 @dblink 追加到表名或视图名来引用另一个数据库上的表或视图。您可以查询另一个数据库上的表或视图，或对该表使用任意 INSERT、UPDATE、DELETE 或 LOCK TABLE 语句。

## 连接到其它数据库

```
REMOTE_ORCL =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)
  (HOST = host02.example.com)
  (PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = orcl.example.com)
  )
)
```

tnsnames.ora

```
CONNECT hr/hr@orcl;

CREATE DATABASE LINK remote
CONNECT TO HR IDENTIFIED BY HR
USING 'REMOTE_ORCL';

SELECT * FROM employees@remote
```

SQL\*Plus

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 连接到其它数据库

此幻灯片显示了在创建数据库链接之前需要的 tnsnames.ora 条目。该示例显示了一个名为 REMOTE 的固定用户数据库链接，它使用连接字符串 REMOTE\_ORCL 连接到用户 HR。创建数据库链接之后，可以使用它来引用另一个数据库上的表和视图。

视图的说明如下所示：

SQL> DESC DBA_DB_LINKS		
Name	Null?	Type
OWNER	NOT NULL	VARCHAR2 (30)
DB_LINK	NOT NULL	VARCHAR2 (128)
USERNAME		VARCHAR2 (30)
HOST		VARCHAR2 (2000)
CREATED	NOT NULL	DATE

```
SQL> select owner, db_link, username from dba_db_links;
```

OWNER	DB_LINK	USERNAME
HR	REMOTE.EXAMPLE.COM	HR

## 小测验

使用哪些配置文件配置监听程序？

1. listener.ora
2. listener.conf
3. tnsnames.ora
4. tnsnames.conf
5. sqlnet.ora
6. sqlnet.conf

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：1 和 5

## 小测验

使用共享服务器进程体系结构时，PGA 重新定位到 SGA 中。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2

## 小结

在本课中，您应该已经学会：

- 使用 Enterprise Manager 执行以下操作：
  - 创建其它监听程序
  - 创建 Oracle Net 服务别名
  - 配置连接时故障转移
  - 控制 Oracle Net 监听程序
- 使用 tnsping 测试 Oracle Net 的连接
- 确定何时使用共享服务器以及何时使用专用服务器

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 6 概览：使用 Oracle Network 组件

本练习包含以下主题：

- 通过配置本地名称解析来连接到其它数据库
- 为连接时故障转移创建第二个监听程序

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



## 管理数据库存储结构

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

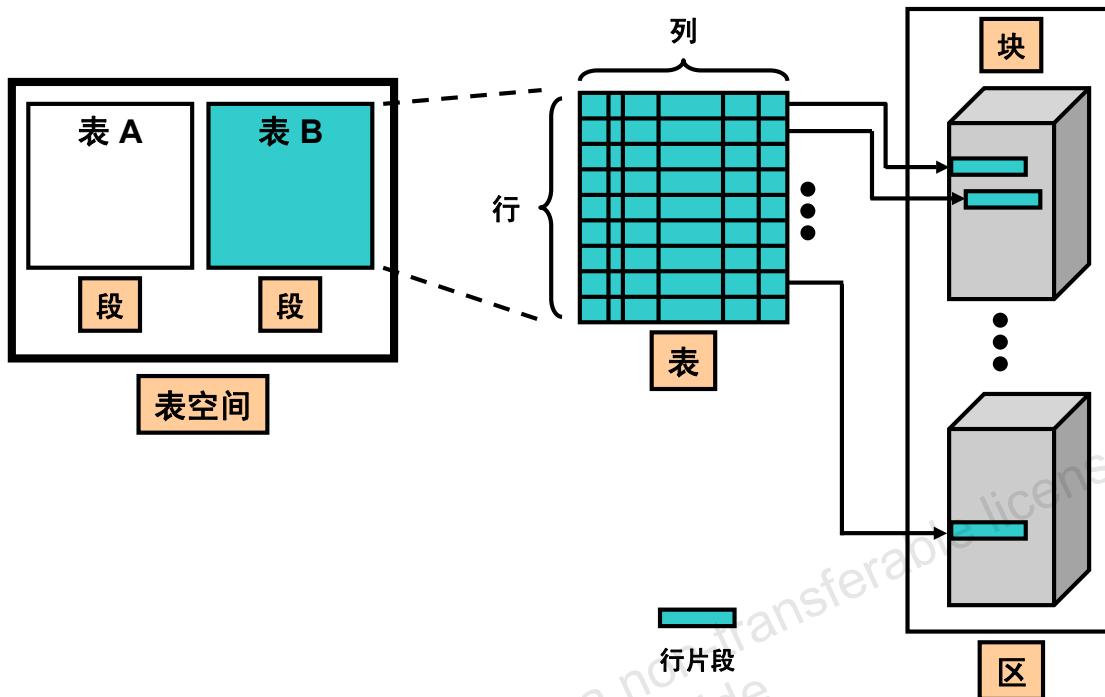
学完本课后，应能完成以下工作：

- 描述块中表行数据的存储
- 创建和管理表空间
- 获取表空间信息

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 表数据的存储方式



ORACLE

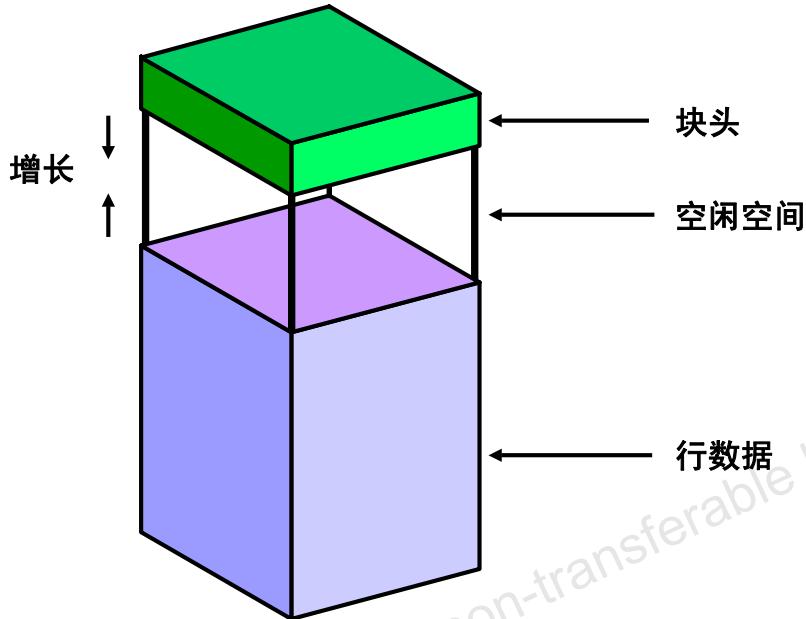
版权所有 © 2010, Oracle。保留所有权利。

### 表数据的存储方式

创建表时，还会创建一个用于保存其数据的段。表空间包含一个段集合。

从逻辑上来说，表包含许多行列值。行最终以行片段的形式存储在数据库块中。之所以称其为行片段，是因为在某些情况下可能不会在一个位置存储完整的行。当插入的行太大而无法装入单个块时（链接行），或当更新导致现有行超出当前块的可用空闲空间时（迁移行），就会发生这种情况。当表中的列多于 255 列时，也会使用行片段。在这种情况下，这些行片段可能位于同一个块（块内链）中，也可能位于多个块中。

## 数据库块：内容



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据库块：内容

- 块头:** 块头包含段类型（如表或索引）、数据块地址、表目录、行目录和事务处理插槽。每个插槽的大小约为 23 字节，修改块中的行时会使用这些插槽。块头自上而下进行增长。
- 行数据:** 这是块中行的实际数据。行数据空间自下而上进行增长。
- 空闲空间:** 空闲空间位于块的中部，允许头和行数据空间在必要时进行增长。当插入新行或用更大的值更新现有行的列时，行数据会占用空闲空间。

导致块头增长的事件的示例包括：

- 行目录需要更多的行条目
- 需要的事务处理插槽数多于最初配置的数目

起初，块中的空闲空间是相邻的。但是，删除和更新操作可能会使块中的空闲空间变成碎片。需要时 Oracle 服务器会接合块中的空闲空间。

## 浏览存储结构

The screenshot shows the Oracle Enterprise Manager 11g interface. At the top, it says "ORACLE Enterprise Manager 11g" and "Database Control". Below that, it says "Database Instance: orcl.oracle.com". There is a navigation bar with tabs: Home, Performance, Availability, **Server**, Schema, and Data Movement. The "Server" tab is selected. On the left, under the "Storage" heading, there is a list of links: Control Files, Tablespaces, Temporary Tablespace Groups, Datafiles, Rollback Segments, Redo Log Groups, Archive Logs, Disk Groups, Migrate to ASM, and Make Tablespace Locally Managed. To the right, under "Database Configuration", there are links: Memory Advisors, Automatic Undo Management, Initialization Parameters, and View Database Feature Usage. At the bottom of the screenshot, there is a red banner with the "ORACLE" logo and the text "版权所有 © 2010, Oracle。保留所有权利。". A blue box with the text "单击链接可查看详细信息。" has an arrow pointing to the "Storage" link in the screenshot.

### 浏览存储结构

逻辑数据结构存储在数据库的物理文件中。通过 Enterprise Manager (EM) 可方便地查看数据库的逻辑结构。通过单击“Server（服务器）”页的“Storage（存储）”区域中的链接，可以获得关于每一种结构的详细信息。

## 创建新的表空间

**Create Tablespace**

**General Storage**

\* Name INVENTORY

**Extent Management**

- Locally Managed
- Dictionary Managed

**Type**

- Permanent
  - Set as default permanent tablespace
  - Encryption [Encryption Options](#)
- Temporary
  - Set as default temporary tablespace
- Undo
  - Undo Retention Guarantee  Yes  No

**Status**

- Read Write
- Read Only
- Offline

**Datafiles**

Use bigfile tablespace  
Tablespace can have only one datafile with no practical size limit.

Select Name	Directory	Add	Size (MB)
No items found			

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 创建新的表空间

- 单击“Server（服务器）”选项卡，然后单击“Storage（存储）”标题下的“Tablespaces（表空间）”。
  - 单击“Create（创建）”。
- 注：**如果要创建与现有表空间类似的表空间，请选择一个现有表空间，然后从“Actions（操作）”菜单中选择“Create Like（类似创建）”。单击“Go（执行）”。此时会出现“Create Tablespace（创建表空间）”页。
- 输入表空间的名称。
  - 在“Extent Management（区管理）”标题下，选择“Locally Managed（本地管理的）”。Oracle DB 服务器可以在表空间中有效地管理本地管理的表空间的区。对于字典管理的表空间，您必须更主动地管理区，而且需要数据字典访问权限才能跟踪这些区。提供“Dictionary-managed（字典管理的）”选项只是为了向后兼容；Oracle 不建议使用此选项。
  - 在“Type（类型）”标题下，选择“Permanent（永久）”。
- 永久表空间用于存储系统或用户创建的永久数据库对象。

## 创建新的表空间（续）

6. 在“Status（状态）”标题下，选择“Read Write（读写）”。  
读写状态表示，在创建表空间后用户可以对表空间进行读写操作。这是默认设置。
7. 在该页的“Datafiles（数据文件）”区域中，单击“Add（添加）”将数据文件添加到该表空间。

**创建新的表空间**

**选择适当的“Storage Type (存储类型)”**

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 创建新的表空间（续）

一个表空间必须至少有一个文件。根据您的环境，选择适当的存储类型。大文件表空间用于超大型数据库，在这些数据库中，ASM 或其它逻辑卷管理器支持条带化或独立磁盘冗余阵列 (RAID)，并且支持可动态扩展的逻辑卷。

8. 在“Add Datafiles (添加数据文件)”页上选择所需的存储类型，并输入必需的信息。对于 ASM，选择所需的磁盘组。对于“File System (文件系统)”，为数据文件输入文件名和文件目录。
9. 输入所需的文件大小。
10. 在“Storage (存储)”区域中，选择“Automatically extend datafile when full (AUTOEXTEND) (数据文件满后自动扩展 (AUTOEXTEND))”，然后在“Increment (增量)”字段中指定一个数量。这样，每次数据文件超出空间后都会自动扩展。当然，数据文件受限于它所驻留的物理介质。将“Maximum File Size (最大文件大小)”保留为“Unlimited (无限制)”或为其输入一个最大值。单击“Continue (继续)”，返回到“Create Tablespace (创建表空间)”页。
11. 返回到“Create Tablespace (创建表空间)”页后，单击“Storage (存储)”选项卡，根据您的需要修改该表空间的存储选项。大多数情况下，接受“Storage (存储)”页上的所有默认值即可。单击“OK (确定)”创建表空间。

## 创建新的表空间（续）

注：这些步骤介绍了大多数情况下快速创建表空间的方法。您可能需要根据具体需求更改某些选项。

## 表空间的存储

**Extent Allocation**

Automatic  
 Uniform  
 Size  KB

**Segment Space Management**

Automatic  
 Objects in the tablespace automatically manage their free space. It offers high performance for free space management.  
 Manual  
 Objects in the tablespace will manage their free space using free lists. It is provided for backward compatibility.

**Compression Options**

Enabling data segment compression can reduce disk usage.  
 Compression  Disabled  
 Enabled on direct-path INSERT operations only  
 Enabled on all operations

**Enable logging**

Yes  
 Generate redo logs for creation of tables, indexes and partitions, and for subsequent inserts. Recoverable  
 No  
 Redo log entries are smaller, the above operations are not logged and not recoverable.

**Block information**

Block Size (B) **8192**

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## 表空间的存储

**区分配:** 可通过以下两种方法之一分配本地管理表空间中的区：

- **自动:** 也称为“自动分配”，这种方法指定表空间中的区的大小是由系统管理的。不能对临时表空间指定“Automatic（自动）”。
- **统一:** 这种方法规定使用您指定的统一区大小来管理表空间。默认大小为 1 MB。临时表空间的所有区都是统一的。不能对还原表空间指定“Uniform（统一）”。

**段空间管理:** 本地管理的表空间中的段空间管理方式可指定为：

- **自动:** Oracle DB 使用位图管理段中的空闲空间。位图描述了段中每个数据块的状态，该状态与可插入行的块中的空间量有关。当数据块中可用空间增多或减少时，位图中会反映数据块的新状态。通过使用位图，Oracle DB 可以提高管理空闲空间的自动化程度。因此，这种空间管理方式称为“自动段空间管理 (ASSM)”。
- **手动:** 此方法指定要使用空闲列表来管理段中的空闲空间。空闲列表是由一些数据块组成的列表，这些数据块中有可插入行的空间。由于这种管理段空间的方式需要为在表空间中创建的方案对象指定并优化 PCTUSED、FREELISTS 和 FREELIST GROUPS 存储参数，因此这种方式称为“手动段空间管理”。支持使用此方法是为了向后兼容，建议使用 ASSM。

## 表空间的存储 (续)

**压缩选项:** 默认情况下, 数据段压缩功能处于禁用状态。启用数据段压缩功能可以节省磁盘空间使用量、减少缓冲区高速缓存中的内存使用量, 还可以加快读取时执行查询的效率。但是, 在数据加载和 DML 过程中会产生 CPU 开销。本功能在联机分析处理 (OLAP) 系统中尤其有用, 在这些系统中存在长时间的只读运算; 但也可用在联机事务处理 (OLTP) 系统中。

有关何时使用压缩子句的详细信息, 请参阅《Oracle 数据库管理员指南》。

**事件记录:** 事件记录子句为表空间中创建的所有段设置默认的事件记录值。对表空间中的对象所做的更改会写入重做日志。如果未启用事件记录, 则使用 SQL\*Loader 和直接加载 INSERT 操作进行的任何直接加载都不会写入重做日志, 因此, 如果发生数据丢失的情况, 这些对象将不可恢复。如果在未启用事件记录的情况下创建了对象, 则只有备份这些对象才能进行恢复。选择不启用事件记录会对日后恢复对象的能力造成巨大影响。请谨慎使用。有关事件记录子句的详细信息, 请参阅《Oracle Database SQL Reference》。

**注:** 如果对数据库启用了 FORCE LOGGING 模式, 该模式的优先级高于表空间事件记录设置。可以在创建数据库时将数据库置于 FORCE LOGGING 模式, 也可以在创建数据库后使用 alter database force logging 命令将数据库置于该模式。

**块信息:** 此区域显示所创建的表空间使用的块大小。此处显示的值为只读值。如果设置了任何其它块大小初始化参数 (DB\_nK\_CACHE\_SIZE), 则会在此处列出其它值作为选项。

有关定义其它块大小的详细信息, 请参阅《Oracle 数据库管理员指南》。

## 预配置的数据库中的表空间

- SYSTEM
- SYSAUX
- TEMP
- UNDOTBS1
- USERS
- EXAMPLE (可选)

**Tablespaces**

Search  
Enter an object name to filter the data that is displayed in your results set.  
Object Name  Go

By default, the search returns all uppercase matches beginning with the string you entered. To run an exact or case-sensitive match, double quote the search string. You can use the wildcard symbol (%) in a double quoted string.

Selection Mode Single

Select	Name	Allocated Size(MB)	Space Used(MB)	Allocated Space Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
<input checked="" type="radio"/>	EXAMPLE	100.0	78.8	78.8 YES		21.2	✓	1	PERMANENT	LOCAL	AUTO
<input type="radio"/>	SYSAUX	697.2	663.9	95.2 YES		33.3	✓	1	PERMANENT	LOCAL	AUTO
<input type="radio"/>	SYSTEM	750.0	744.2	99.2 YES		5.8	✓	1	PERMANENT	LOCAL	MANUAL
<input type="radio"/>	TEMP	27.0	0.0	0.0 YES		27.0	✓	1	TEMPORARY	LOCAL	MANUAL
<input type="radio"/>	UNDOTBS1	100.0	16.1	16.1 YES		83.9	✓	1	UNDO	LOCAL	MANUAL
<input type="radio"/>	USERS	5.0	4.1	82.5 YES		0.9	✓	1	PERMANENT	LOCAL	AUTO

Total Allocated Size (GB) 1.64      ✓ Online    ✘ Offline    ⚡ Read Only  
 Total Used (GB) 1.47  
 Total Allocated Free Space (GB) 0.17

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 预配置的数据库中的表空间

在本课中，在预配置的数据库中创建了以下表空间：

• **SYSTEM:** Oracle 服务器使用 SYSTEM 表空间管理数据库。这个表空间包含的数据字典和表中包含关于数据库的管理信息。上述信息均包含在 SYS 方案中，只有 SYS 用户或者拥有所需权限的其它管理用户才可访问这些信息。

• **SYSAUX:** 这是 SYSTEM 表空间的辅助表空间。Oracle DB 早期版本中某些使用 SYSTEM 表空间或其本身表空间的组件和产品现在改为使用 SYSAUX 表空间。每个 Oracle Database 10g (或更高版本) 数据库都必须拥有 SYSAUX 表空间。

在 Enterprise Manager (EM) 中，可以查看这个表空间中内容的饼图。为此，请单击“Administration (管理)”页上的“Tablespaces (表空间)”。选择“SYSAUX”，单击“Edit (编辑)”。然后单击“Occupants (占有者)”选项卡。完成创建后，可以使用 EM 来监视 SYSAUX 表空间中的每个占有者的空间使用情况。如果检测到某个组件占用了 SYSAUX 表空间中过多的空间，或者如果预计该组件会占用过多的空间，则可以通过选择其中一个占有者，然后单击“Change Tablespace (更改表空间)”，将选择的占有者移到另外的表空间。

## 预配置的数据库中的表空间（续）

- **TEMP:** 如果执行的 SQL 语句需要创建临时段（如大规模排序或创建索引），则需要使用临时表空间。如同为了存储所创建的数据对象而向每一用户分配默认表空间一样，还会向每一用户分配一个临时表空间。最好的做法是为数据库定义一个默认临时表空间，并且将此表空间分配给所有新建用户（除非另行指定）。在预配置的数据库中，TEMP 表空间被指定为默认临时表空间。这表示，如果在创建用户帐户时未指定临时表空间，则 Oracle DB 会将此表空间分配给用户作为临时表空间。
- **UNDOTBS1:** 这是数据库服务器用于存储还原信息的还原表空间。如果数据库使用“自动还原管理”，那么数据库在任何指定时间只能使用一个还原表空间。此表空间是在创建数据库时创建的。
- **USERS:** 此表空间用于存储用户对象和数据。如果在创建用户时未指定默认的表空间，则 USERS 表空间将成为该用户创建的所有对象的默认表空间。对于 SYS 和 SYSTEM 用户，默认的永久表空间是 SYSTEM。
- **EXAMPLE:** 此表空间包含创建数据库时可以安装的示例方案。这些示例方案为各种示例提供了一个通用平台。在 Oracle 文档和课件中包含了建立在这些示例方案基础上的示例。

注：为简化管理，通常为索引使用单独的表空间。

## 变更表空间

The screenshot shows the Oracle Database 11g Control Panel. In the top navigation bar, 'Edit', 'View', 'Delete', 'Actions', 'Add Datafile', and 'Go' buttons are visible. Below the navigation bar is a table with columns: Select Name /, Allocated Size(MB), Space Used(MB), Allocated Space Used(%), Auto Extend, Allocated Free Space(MB), Status, Datafiles, Type, Extent Management, and Segment Management. Two rows are listed: 'EXAMPLE' (selected) and 'SYSAUX'. The 'EXAMPLE' row has values: 100.0, 78.8, 78.8 YES, 21.2, green checkmark, 1 PERMANENT, LOCAL, AUTO. The 'SYSAUX' row has values: 0, 0, 0, NO, 0, red X, 0, TEMPORAL, SYSTEM, SYSTEM. A red box highlights the 'Select Name /' dropdown for 'EXAMPLE'. An arrow points from this box to the 'Extent Management' section of the 'Edit Tablespace: EXAMPLE' dialog. The 'Extent Management' section shows 'Locally Managed' selected. The 'Type' section shows 'Permanent' selected. The 'Status' section shows 'Read Write' selected. The 'Datafiles' section lists one datafile named 'example.265.688820635'.

### 变更表空间

创建表空间后，可以根据系统更改的需要，采用若干种方法来变更表空间。

**重命名：**输入表空间的新名称，然后单击“Apply（应用）”。

**更改状态：**表空间有三种不同状态，它可以处于其中一种状态。由于表空间的可用性取决于表空间的类型，因此以下三种状态不一定可用：

- 读写：**表空间已联机，可进行读写。
- 只读：**指定“只读”可将表空间置于只读模式。此状态下，可以完成（提交或回退）现有的事务处理，但是，不允许对表空间中的对象进一步执行数据操纵语言（DML）操作。表空间已联机，但处于只读状态。不能使 SYSTEM 或 SYSAUX 表空间处于只读模式。

**注：**无法使还原表空间和临时表空间成为只读表空间。

## 变更表空间（续）

- **脱机**：您可以让联机的表空间脱机，使数据库的这部分暂时不可用于一般用途。数据库的余下部分是开放的，可让用户访问其中的数据。表空间脱机时，可以使用以下选项：
  - **Normal (正常)**：如果表空间中的任一数据文件都不存在任何错误状态，通过正常方式便可使表空间脱机。当 Oracle DB 使表空间脱机时，通过为表空间的所有数据文件设置检查点，来确保将所有数据写入磁盘。
  - **Temporary (临时)**：如果表空间中的一个或多个文件存在错误状态，也可以使表空间暂时脱机。当 Oracle DB 使数据文件（尚未脱机的）脱机时，将为这些数据文件设置检查点。如果没有任何文件脱机，但是您使用了临时子句，则在使表空间重新联机时不需要执行介质恢复。但是，如果因写错误而导致表空间的一个或多个文件脱机，而且设置了表空间临时脱机，那么表空间需要执行恢复后才能重新联机。
  - **Immediate (立即)**：Oracle DB 可以使表空间立即脱机，而不需要为任何数据文件设置检查点。如果指定了“**Immediate (立即)**”，则必须先对表空间执行介质恢复，才能使表空间联机。如果数据库在 NOARCHIVELOG 模式下运行，则无法立即使表空间脱机。
  - **For Recover (用于恢复)**：FOR RECOVER 设置已被废弃。支持此语法是为了向后兼容。

**注：**不能使系统表空间脱机。

**更改大小：**通过向表空间添加数据文件，或者更改现有数据文件的大小，可增加现有表空间的空间。

- 要向表空间添加新的数据文件，请单击“**Add (添加)**”，然后在“**Add Datafile (添加数据文件)**”页中输入有关数据文件的信息。  
**注：**无法向大文件表空间中继续添加其它数据文件。
- 要更改现有数据文件的大小，请单击数据文件的名称，在“**Edit Tablespace (编辑表空间)**”页的“**Datafiles (数据文件)**”区域中选择数据文件；或者选择数据文件，然后单击“**Edit (编辑)**”。接下来，在“**Edit Datafile (编辑数据文件)**”页上，可更改数据文件的大小。可以扩大或缩小表空间。但是，不能使数据文件小于该文件中已使用的空间；如果尝试这样做，则会出现以下错误：

```
ORA-03297: file contains used data beyond requested RESIZE value
```

**存储选项：**单击“**Storage (存储)**”可更改表空间的事件记录行为。

**阈值：**单击“**Thresholds (阈值)**”可更改表空间的空间使用到达警告或严重级别时的值。共有三个选项：

- **Use Database Default Thresholds (使用数据库默认阈值)**：使用预设的默认值，而且提供用于设置这些默认值的选项。
- **Specify Thresholds (指定阈值)**：可以设置这个特定表空间的阈值。
- **Disable Thresholds (禁用阈值)**：关闭这个表空间的空间使用预警。

**注：**默认情况下，每隔 10 分钟才检查一次空间使用率，因此，可能要花费数分钟时间才会注册一条阈值预警。

## 表空间操作

The screenshot shows the Oracle Database 11g Tablespace Management interface. A red arrow points from the 'Actions' menu in the top navigation bar down to the 'Generate DDL' option in the dropdown menu. The 'EXAMPLE' tablespace is selected in the list. The table below lists various tablespaces with their properties like Allocated Size (MB), Auto Extend, and Status. A summary at the bottom provides total allocated size, used space, and free space.

Select	Name	Allocated Size(MB)	Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
<input checked="" type="radio"/>	EXAMPLE	100.0	78.8	YES	21.2	✓	1	PERMANENT LOCAL	AUTO	
<input type="radio"/>	SYSAUX	697.2	95.2	YES	33.3	✓	1	PERMANENT LOCAL	AUTO	
<input type="radio"/>	SYSTEM	750.0	99.2	YES	5.8	✓	1	PERMANENT LOCAL	MANUAL	
<input type="radio"/>	TEMP	27.0	0.0	YES	27.0	✓	1	TEMPORARY LOCAL	MANUAL	
<input type="radio"/>	UNDOTBS1	100.0	17.1	YES	82.9	✓	1	UNDO LOCAL	MANUAL	
<input type="radio"/>	USERS	5.0	4.1	YES	0.9	✓	1	PERMANENT LOCAL	AUTO	

Total Allocated Size (GB) 1.64      ✓ Online    ✘ Offline    ⚡ Read Only  
Total Used (GB) 1.47  
Total Allocated Free Space (GB) 0.17

**Show DDL**

```
CREATE SMALLFILE TABLESPACE "EXAMPLE" DATAFILE '+DATA/orcl/datafile/example.265.688820635'
SIZE 100M REUSE AUTOEXTEND ON NEXT 640K MAXSIZE 32767M NOLOGGING EXTENT MANAGEMENT LOCAL
SEGMENT SPACE MANAGEMENT AUTO
```

[Return](#)

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 表空间操作

使用“Actions（操作）”菜单，可以对表空间执行各种任务。选择一个表空间，然后选择要执行的操作。

- Add Datafile (添加数据文件)**：将数据文件添加到表空间，这会增大表空间。
- Create Like (类似创建)**：将表空间用作模板，创建另一个表空间。
- Generate DDL (生成 DDL)**：生成用于创建表空间的数据定义语言 (DDL) 语句。随后，此语句可复制并粘贴到文本文件，以用作脚本或文档。
- Make Locally Managed (进行本地管理)**：如果表空间当前是字典管理的表空间，则可将该表空间转换为本地管理的表空间。这种转换只能单向进行；不能将表空间重新转换到字典管理的表空间。如果需要，可以使用 PL/SQL 程序包 DBMS\_SPACE\_ADMIN.TABLESPACE\_MIGRATE\_FROM\_LOCAL 来转换到字典管理的表空间。
- Make Readonly (只读)**：停止对表空间的所有写操作。允许完成当前的事务处理，但是，不允许对表空间启动新的 DML 活动或其它写活动。只有表空间的当前状态不是只读时，才显示此选项。
- Make Writable (可写)**：允许对表空间中的对象启动 DML 活动和其它写活动。只有表空间的当前状态不是可写时，才显示此选项。

## 表空间操作（续）

- **Place Online (联机)**：使当前脱机的表空间联机。
- **Reorganize (重组)**：启动“Reorganization Wizard (重组向导)”，使用此向导可在表空间中各处移动对象以回收空间，使其变为可用。这个任务应当在表空间中的对象处于非使用高峰时执行。
- **Run Segment Advisor (运行段指导)**：启动“Segment Advisor (段指导)”，使用该指导可根据对象中空间碎片的级别来确定对象是否拥有可回收的空间。会在表空间级别上针对表空间中的每个段生成建议。
- **Show Dependencies (显示依赖性)**：显示此表空间依赖的对象，或依赖于此表空间的对象。
- **Show Tablespace Contents (显示表空间内容)**：显示关于表空间中所有段的信息，包括所有区的图形。
- **Take Offline (脱机)**：使当前联机的表空间不可用。此时不会删除表空间，只是表空间不可用。

## 删除表空间

**⚠ Warning**

Once a tablespace has been dropped, the objects and data in it will no longer be available. To recover them can be a time consuming process. Oracle recommends a backup before and after dropping a tablespace.

Are you sure you want to delete Tablespace EXAMPLE?

Delete associated datafiles from storage

Select	Name	Allocated Size(MB)	Space Used(MB)	Allocated Space Used(%)	Auto Extend	Allocated Free Space(MB)	Status	Datafiles	Type	Extent Management	Segment Management
<input checked="" type="radio"/>	EXAMPLE	100.0	78.8	78.8 YES	21.2	✓	1	PERMANENT	LOCAL	AUTO	
<input type="radio"/>	SYSAUX	697.2	663.9	95.2 YES	33.3	✓	1	PERMANENT	LOCAL	AUTO	
<input type="radio"/>	SYSTEM	750.0	744.2	99.2 YES	5.8	✓	1	PERMANENT	LOCAL	MANUAL	
<input type="radio"/>	TEMP	27.0	0.0	0.0 YES	27.0	✓	1	TEMPORARY	LOCAL	MANUAL	
<input type="radio"/>	UNDOTBS1	100.0	17.1	17.1 YES	82.9	✓	1	UNDO	LOCAL	MANUAL	
<input type="radio"/>	USERS	5.0	4.1	82.5 YES	0.9	✓	1	PERMANENT	LOCAL	AUTO	

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 删除表空间

如果不再需要表空间及其内容（表空间中包含的段），可以从数据库中删除表空间及其内容。必须具有 DROP TABLESPACE 系统权限才可以删除表空间。

删除表空间时，会删除关联数据库控制文件中的文件指针。如果在使用 Oracle 管理的文件 (OMF)，则还会删除基础操作系统文件。如果未使用 OMF，则可以根据需要指示 Oracle 服务器删除包含在已删除表空间中的操作系统文件（数据文件）。如果不指示 Oracle 服务器在删除表空间的同时删除数据文件，那么，当您以后要删除这些文件时，必须使用操作系统的相应命令。

不能删除包含活动段的表空间。例如，如果某个表空间中的某个表当前正在使用中，或者该表空间包含回退未提交事务处理所需的还原数据，则不能删除该表空间。表空间的状态可以是联机的也可以是脱机的，但是在删除表空间之前，最好使其处于脱机状态。

## 查看表空间信息

```
SELECT tablespace_name, status, contents, logging, extent_management,
allocation_type, segment_space_management
FROM dba_tablespaces
```

TABLESPACE_NAME	STATUS	CONTENTS	LOGGING	EXTENT_MANAGEMENT	ALLOCATION_TYPE	SEGMENT_SPACE_MANAGEMENT
SYSTEM	ONLINE	PERMANENT	LOGGING	LOCAL	SYSTEM	MANUAL
SYSAUX	ONLINE	PERMANENT	LOGGING	LOCAL	SYSTEM	AUTO
UNDOTBS1	ONLINE	UNDO	LOGGING	LOCAL	SYSTEM	MANUAL
TEMP	ONLINE	TEMPORARY	NOLOGGING	LOCAL	UNIFORM	MANUAL
USERS	ONLINE	PERMANENT	LOGGING	LOCAL	SYSTEM	AUTO
EXAMPLE	ONLINE	PERMANENT	NOLOGGING	LOCAL	SYSTEM	AUTO

```
SELECT ts#, name FROM v$tablespace
```

TS#	NAME
0	SYSTEM
1	SYSAUX
2	UNDOTBS1
4	USERS
3	TEMP
6	EXAMPLE

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 查看表空间信息

单击“View（查看）”可查看关于选定表空间的信息。在“View Tablespace（查看表空间）”页上，还可通过单击“Edit（编辑）”来变更表空间。

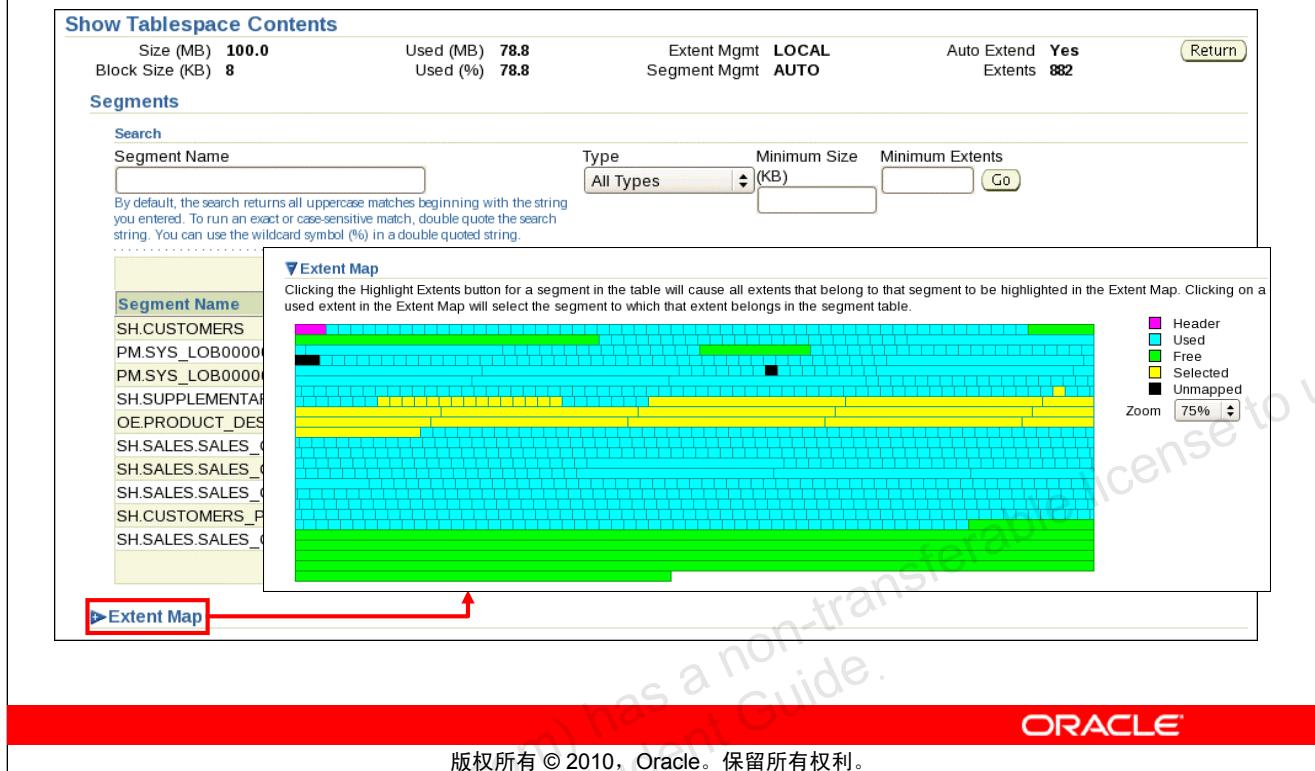
通过查询以下项也可以获取表空间和数据文件信息：

- 表空间信息：
  - DBA\_TABLESPACES
  - V\$TABLESPACE
- 数据文件信息：
  - DBA\_DATA\_FILES
  - V\$DATAFILE

注：V\$DBFILE 视图显示数据库中的所有数据文件。保留此视图是为了向后兼容。建议改用 V\$DATAFILE。

- 临时文件信息：
  - DBA\_TEMP\_FILES
  - V\$TEMPFILE

## 查看表空间内容



## 查看表空间内容

在“Tablespaces（表空间）”主页或特定表空间页上，从“Actions（操作）”下拉列表中选择“Show Tablespace Contents（显示表空间内容）”，然后单击“Go（执行）”。

“Show Tablespace Contents（显示表空间内容）”页中显示了关于表空间的详细信息，其中包括表空间中段的列表、每个段的类型、段大小以及每个段中的区数。使用这四个值中的任何一个值，通过单击列标题可对列表进行排序，或者通过在“Search（搜索）”区域中输入值对列表进行过滤。对于字典管理的表空间，将显示以下额外的列：

- Max Extents（最大区数）
- Next（下一个）
- Percent Increase（增加百分比）

要查看区列表，请单击“Extents（区）”列中的链接。

### 查看表空间内容（续）

要以图形方式查看区，请展开“Extent map（区映射）”，然后在各个区上移动光标。此时会显示以下信息：

- 区所属的段的名称
- 区 ID
- 块 ID
- 区大小（以块数计）
- 用来存储区的数据文件

## Oracle 管理的文件 (OMF)

按照数据库对象而不是文件名指定文件操作。

参数	说明
DB_CREATE_FILE_DEST	定义数据文件和临时文件默认文件系统目录的位置
DB_CREATE_ONLINE_LOG_DEST_n	定义重做日志文件和控制文件的创建位置
DB_RECOVERY_FILE_DEST	快速恢复区的默认位置

示例：

```
SQL> ALTER SYSTEM SET DB_CREATE_FILE_DEST = '+DATA';
SQL> CREATE TABLESPACE tbs_1;
```

版权所有 © 2010, Oracle。保留所有权利。

### Oracle 管理的文件 (OMF)

如果使用 Oracle 管理的文件，则不需要直接管理 Oracle DB 中的操作系统文件。可按照数据库对象而不是文件名指定操作。数据库将根据需要，在内部使用标准文件系统接口创建或删除下列数据库结构的文件：

- 表空间
- 重做日志文件
- 控制文件
- 归档日志
- 块更改跟踪文件
- 闪回日志
- RMAN 备份

数据库既可以包含 Oracle 管理的文件，也可以包含非 Oracle 管理的文件。由这两个参数之一指定的文件系统目录必须都已存在；数据库不会创建该目录。该目录还必须具有相应的权限，以便数据库在其中创建文件。

示例说明在设置 DB\_CREATE\_FILE\_DEST 后，可以省略 CREATE TABLESPACE 语句中的 DATAFILE 子句。将在 DB\_CREATE\_FILE\_DEST 指定的位置创建数据文件。按所示方式创建表空间时，会为所有参数分配默认值。

## Oracle 管理的文件 (OMF) (续)

Oracle 管理的文件具有特定的命名格式。例如，在基于 Linux 和 UNIX 的系统中使用以下格式：

```
<destination_prefix>/o1_mf_%t_%u_.dbf
```

请勿重命名 Oracle 管理的文件。数据库通过名称来识别 Oracle 管理的文件。重命名文件会导致数据库无法将其识别为 Oracle 管理的文件，从而无法正确地管理该文件。

以下示例将创建数据文件的默认位置设置为 /u01/oradata，然后创建了一个在该位置存储数据文件的表空间 tbs\_1。

```
SQL> ALTER SYSTEM SET DB_CREATE_FILE_DEST = '/u01/oradata';
SQL> CREATE TABLESPACE tbs_1;
```

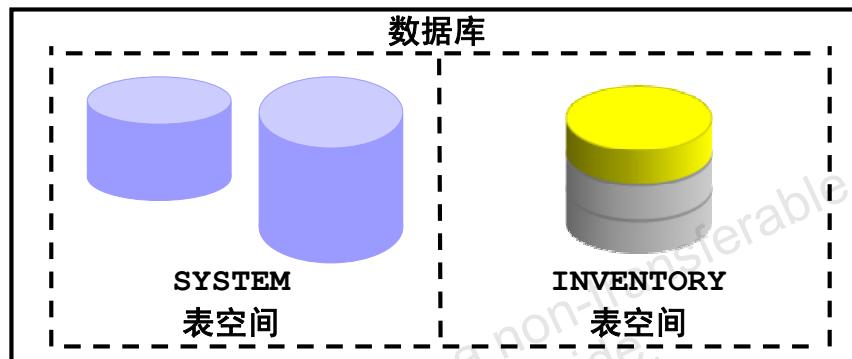
默认情况下，Oracle 管理的数据文件（包括 SYSTEM 和 SYSAUX 表空间中 Oracle 管理的数据文件）为 100MB 大小且可以自动扩展。

**注：**默认情况下，ASM 使用 OMF 文件，但是，如果在创建表空间时或向现有表空间中添加 ASM 数据文件时为 ASM 数据文件指定了别名，则该 ASM 数据文件不会为 OMF 格式。

## 扩大数据库

可以按以下方式扩大数据库：

- 创建新的表空间
- 将数据文件添加到现有的小文件表空间
- 增加数据文件的大小
- 动态扩展数据文件



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 扩大数据库

可以使用 Enterprise Manager 或 SQL 语句执行这些活动。可以将数据库大小描述为数据库的所有表空间的大小之和。

## 小测验

数据库既可以包含 Oracle 管理的文件，也可以包含非 Oracle 管理的文件。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 1

## 小测验

一个表空间必须至少有一个 100 MB 大小的文件。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小结

在本课中，您应该已经学会：

- 描述块中表行数据的存储
- 创建和管理表空间
- 获取表空间信息

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 7 概览：管理数据库存储结构

本练习包含以下主题：

- 创建表空间
- 采集关于表空间的信息

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 管理用户安全性

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 创建和管理数据库用户帐户：
  - 验证用户
  - 分配默认存储区（表空间）
- 授予和撤销权限
- 创建和管理角色
- 创建和管理概要文件：
  - 实施标准口令安全功能
  - 控制用户的资源使用量

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 课程目标

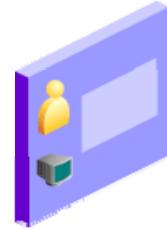
以下术语与管理数据库用户相关，通过了解这些术语可帮助了解课程目标：

- “数据库用户帐户”是一种组织数据库对象的所有权和访问权限的方法。
- “口令”是 Oracle DB 使用的一种验证方法。
- “权限”是用于执行特定类型的 SQL 语句或访问其他用户的对象的一种权限。
- “角色”是一个由相关权限组成的指定组，可授予给用户或其它角色。
- “概要文件”是用于限制数据库使用和实例资源的一组指定资源限制条件，并管理帐户状态和口令管理规则。
- “限额”是允许给定表空间具有的空间。通过这种方法，可以控制用户的资源使用量。

## 数据库用户帐户

每个数据库用户帐户都包括以下项:

- 唯一的用户名
- 验证方法
- 默认表空间
- 临时表空间
- 用户概要文件
- 初始使用者组
- 帐户状态



方案:

- 是数据库用户拥有的数据库对象的集合
- 与用户帐户具有相同的名称

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 数据库用户帐户

要访问数据库，用户必须指定有效的数据库用户帐户，而且还要根据该用户帐户的要求成功通过验证。每个数据库用户都有一个唯一的数据库帐户。Oracle 建议这样做是为了避免潜在的安全漏洞以及为特定的审计活动提供有意义的数据。但是，有时候若干用户会共享一个公用数据库帐户。在这些罕见的情况下，操作系统和应用程序必须为数据库提供足够的安全性。每个用户帐户都包括以下项:

- **唯一的用户名:** 用户名不能超过 30 个字节，不能包含特殊字符，而且必须以字母开头。
- **验证方法:** 最常见的验证方法是口令，但是 Oracle Database 11g 支持口令、全局和外部验证方法（例如生物统计学验证、证书验证和标记验证）。
- **默认表空间:** 如果用户未指定其它表空间，则可在这个位置创建对象。请注意，具有默认表空间并不意味着用户在该表空间具有创建对象的权限，也不意味着用户在该表空间中具有用于创建对象的空间限额。这两项需要另外授权。
- **临时表空间:** 这是实例代表用户创建临时对象（如排序和临时表）的位置。临时表空间没有限额。
- **用户概要文件:** 分配给用户的一组资源与口令限制。

## 数据库用户帐户（续）

- **初始使用者组：**由资源管理器使用。
- **帐户状态：** 用户只可访问“打开”的帐户。`account_status` 可能是“锁定”和“过期”的各种组合。

**方案：**“方案”是数据库用户拥有的数据库对象的集合。方案对象是直接引用数据库数据的逻辑结构。方案对象包括表、视图、序列、存储过程、同义词、索引、集群和数据库链接等结构。通常，方案对象包括应用程序在数据库中创建的所有内容。

**注：** 数据库用户不一定是人员。常见的作法是创建一个拥有特定应用程序的数据库对象的用户，例如 `HR`。数据库用户可以是设备、应用程序或只是一种出于安全目的而对数据库对象进行分组的方法。数据库用户不需要具有个人身份信息。

## 预定义管理帐户

- **SYS 帐户:**
  - 被授予 DBA 角色以及几个其它角色
  - 具有带 ADMIN OPTION 的所有权限
  - 执行启动、关闭和某些维护命令时需要使用该帐户
  - 拥有数据字典和自动工作量资料档案库 (AWR)
- **SYSTEM 帐户被授予 DBA、MGMT\_USER 和 AQ\_ADMINISTRATOR\_ROLE 角色。**
- **DBSNMP 帐户被授予 OEM\_MONITOR 角色。**
- **SYSMAN 帐户被授予 MGMT\_USER、RESOURCE 和 SELECT\_CATALOG\_ROLE 角色。**
- **常规操作不使用这两个帐户。**



版权所有 © 2010, Oracle。保留所有权利。

### 预定义管理帐户

默认情况下，SYS 和 SYSTEM 帐户被授予数据库管理员 (DBA) 角色。另外，SYS 帐户还具有带 ADMIN OPTION 的所有权限并拥有数据字典。要连接到 SYS 帐户，对于数据库实例，必须使用 AS SYSDBA 子句，对于自动存储管理 (ASM) 实例，必须使用 AS SYSASM 子句。授予了 SYSDBA 权限的任何用户都可以通过使用 AS SYSDBA 子句连接到 SYS 帐户。仅允许授予了 SYSDBA、SYSOPER 或 SYSASM 权限的“特权”用户启动和关闭实例。SYSTEM 帐户不具有 SYSDBA 权限。SYSTEM 还被授予 AQ\_ADMINISTRATOR\_ROLE 和 MGMT\_USER 角色。SYS 和 SYSTEM 帐户是数据库所必需的帐户。不能将其删除。

Oracle Enterprise Manager 的管理代理使用 DBSNMP 帐户来监视和管理数据库。SYSMAN 帐户用于执行 Oracle Enterprise Manager 管理任务。DBSNMP 或 SYSMAN 都没有 SYSDBA 权限。

**最佳实践提示：**如果应用最少权限原则，则常规操作不应使用这些帐户。为需要 DBA 权限的用户分配授予了所需权限的单独帐户。例如，Jim 有一个名为 jim 的低权限帐户及一个名为 jim\_dba 的授权帐户。使用此方法可保证最少权限原则，不需要帐户共享，而且可审计各项操作。

## 创建用户

Database Instance: orcl.oracle.com > Users > Logged in As SYS

Create User

General Roles System Privileges Object Privileges Quotas Consumer Group Privileges Proxy Users

\* Name mydba  
Profile DEFAULT  
Authentication Password  
\* Enter Password \*\*\*\*\*  
\* Confirm Password \*\*\*\*\*  
For Password choice, the role is authorized via password.  
 Expire Password now  
Default Tablespace USERS  
Temporary Tablespace TEMP  
Status  Locked  Unlocked

Show SQL

```
CREATE USER "MYDBA" PROFILE "DEFAULT" IDENTIFIED BY "*****" DEFAULT TABLESPACE "USERS" TEMPORARY TABLESPACE "TEMP" ACCOUNT UNLOCK GRANT "CONNECT" TO "MYDBA"
```

选择“Server > Users（服务器 > 用户）”，然后单击“Create（创建）”按钮。

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 创建用户

在 Oracle Enterprise Manager 的“Users（用户）”页中，可以管理能够访问当前数据库的数据库用户的列表。使用此页可创建、删除和修改用户的设置。

要创建数据库用户，请执行以下操作：

1. 在 Enterprise Manager Database Control 中，单击“Server（服务器）”选项卡，然后在“Security（安全）”部分中单击“Users（用户）”。
2. 单击“Create（创建）”按钮。

提供必需的信息。必需项带有星号(\*)标记，如“Name（名称）”。指定的名称必须符合创建数据库对象所用的那些规则。本课中的后面几页提供关于验证的详细信息。本课稍后将介绍概要文件。

请给每一个用户分配默认表空间和临时表空间。如果用户在创建对象时未指定表空间，则将在分配给对象所有者的默认表空间中创建对象。这样，您便可以控制用户对象的创建位置。如果未选择默认表空间，则使用系统定义的默认永久表空间。对于临时表空间也是如此：如果未指定表空间，则使用系统定义的临时表空间。

**注：**单击“Show SQL（显示 SQL）”以查看 SQL 支持语法。有关创建用户的完整 SQL 语法，请参阅《Oracle® Database SQL Language Reference》手册。

## 验证用户

- 口令验证
- 外部验证
- 全局验证

**Edit User: HR**

Actions: Create Like, Go, Show SQL, Revert, Apply

General Roles System Privileges Object Privileges Quotas Consumer Group Privileges Proxy Users

Name: HR  
Profile: DEFAULT  
Authentication: Password (dropdown menu open)  
\* Enter Password: (text input field)  
\* Confirm Password: (text input field)  
For Password choice, the role is authorized via password.  
 Expire Password now  
Default Tablespace: USERS  
Temporary Tablespace: TEMP  
Status:  Locked  Unlocked

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 验证用户

“验证”表示验证要使用数据库、资源或应用程序的某人或某事物（用户、设备或其它实体）的身份。通过对该身份进行验证可建立一种信任关系，以进行进一步的交互。通过验证可将访问和操作与特定的身份联系起来，从而增强安全性。完成验证后，授权进程会允许或限制该实体能够执行的访问和操作的级别。

创建用户时，必须确定要使用的验证方法，以后可修改此方法。

**口令验证：**又称为 Oracle DB 验证。创建的每一个用户都有一个关联口令，用户尝试建立连接时，必须提供这个口令。设置口令时，可以使该口令立即失效，这会强制用户在首次登录后更改口令。如果决定要使用用户口令失效，请确保用户能够更改口令。有些应用程序不具备此功能。默认情况下，在 Oracle Database 11g 中创建的所有口令都区分大小写。

这些口令还可以包含多字节字符，但长度被限制为 30 字节。在升级到 Oracle Database 11g 的数据库中创建的每个口令仍然不区分大小写，直到更改该口令。

在网络（客户机/服务器和服务器/服务器）连接工作期间，系统总是先使用高级加密标准 (AES) 算法以自动且透明的方式对口令进行加密，然后再通过网络发送这些口令。

## 验证用户（续）

**外部验证：**这是使用数据库外部（操作系统、Kerberos 或 Radius）的方法执行的验证。Kerberos 或 Radius 需要使用高级安全选件。用户可以在不指定用户名或口令的情况下连接到 Oracle DB。借助于执行严格验证的高级安全选件，系统可以通过使用生物统计学、x509 证书和标记设备来识别用户。使用外部验证时，数据库依赖于基础操作系统、网络验证服务或外部验证服务来限制对数据库帐户的访问。对于此类登录不会使用数据库口令。如果操作系统或网络服务允许的话，可以使用外部验证来验证用户。如果使用操作系统验证，请设置 OS\_AUTHENT\_PREFIX 初始化参数，并在 Oracle 用户名中使用该前缀。OS\_AUTHENT\_PREFIX 参数定义了一个前缀，Oracle DB 会在每个用户的操作系统帐户名之前添加此前缀。为了实现与 Oracle 软件早期版本的向后兼容，此参数的默认值为 OPS\$。用户尝试建立连接时，Oracle DB 会将带前缀的用户名与数据库中的 Oracle 用户名进行比较。例如，假设 OS\_AUTHENT\_PREFIX 设置如下：

```
OS_AUTHENT_PREFIX=OPS$
```

如果某个用户的操作系统帐户名为 tsmith，该用户需要连接到 Oracle DB 并且由操作系统进行验证，则 Oracle DB 会检查是否存在对应的数据库用户 OPS\$tsmith，如果存在这样的用户，则允许该用户建立连接。对由操作系统验证的用户的引用必须包括前缀，如 OPS\$tsmith 所示。

**注：**OS\_AUTHENT\_PREFIX 初始化参数的文本在某些操作系统中区分大小写。有关此初始化参数的详细信息，请参阅特定于操作系统的 Oracle 文档。

**全局验证：**使用 Oracle 高级安全选件时，可使用 Oracle Internet Directory 通过全局验证来识别用户。

有关高级验证方法的详细信息，请参阅“Oracle DB 安全性”课程。

## 管理员验证

### 操作系统安全性:

- DBA 必须具有创建或删除文件的操作系统权限。
- 普通数据库用户不应具有创建或删除数据库文件的操作系统权限。

### 管理员安全性:

- 对于 SYSDBA、SYSOPER、和 SYSASM 连接:
  - 对于口令文件和严格验证方法，按名称审计 DBA 用户
  - 对于操作系统验证，审计操作系统帐户名
  - 对于授权用户，操作系统验证优先于口令文件验证
  - 口令文件使用区分大小写的口令



版权所有 © 2010, Oracle。保留所有权利。

### 管理员验证

**操作系统安全性:** 在 UNIX 和 Linux 中，DBA 默认情况下属于 oinstall 操作系统组，该组具有创建和删除数据库文件所需的权限。

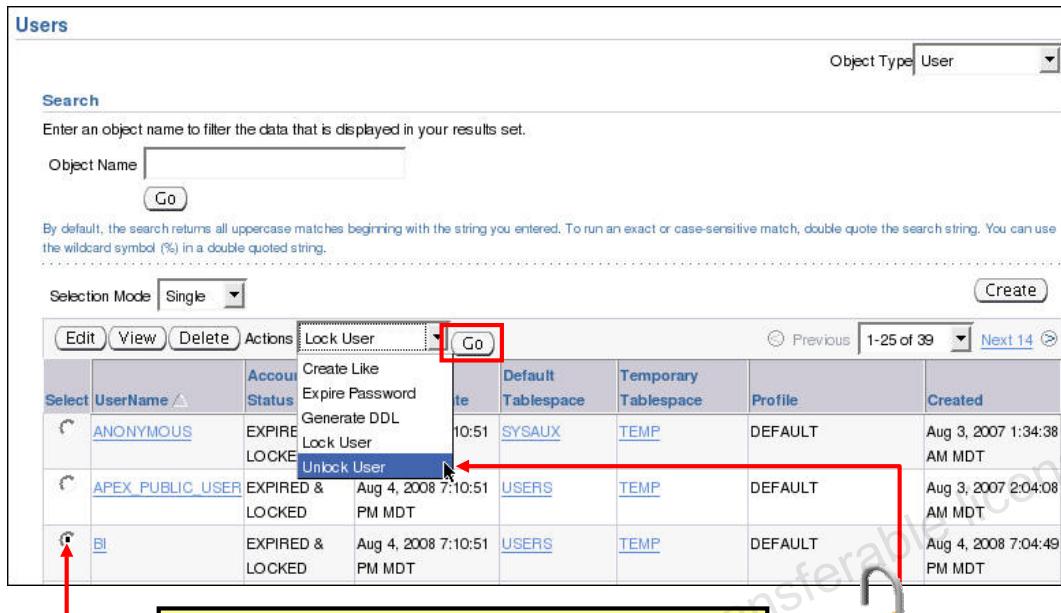
**管理员安全性:** 只有在使用口令文件或操作系统权限进行验证后，才允许授权用户 SYSDBA、SYSOPER 和 SYSASM 建立连接。如果使用操作系统验证，则数据库不使用提供的用户名和口令。当口令文件不存在，或者该文件中不存在提供的用户名和口令，或者未提供用户名和口令时，将使用操作系统验证。默认情况下，Oracle Database 11g 中的口令文件使用区分大小写的口令。

但是，如果使用口令文件成功完成了验证，则使用用户名记录连接。如果使用操作系统成功完成了验证，则表示这是一个 CONNECT / 连接，这种连接不记录具体用户。

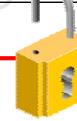
**注:** 如果您是操作系统中 OSDBA 或 OSOPER 组的成员，而且以 SYSDBA 或 SYSOPER 身份进行连接，则会使用关联的管理权限为您建立连接，不管您指定的用户名和口令是什么。对于 SYSASM，您不能指定任何用户名或口令（例如 sqlplus / as SYSASM）。

在 Oracle Database 11g 中，授权用户可以使用严格验证方法：Kerberos、SSL 或目录验证（如果可以使用高级安全选件）。

## 解除用户帐户的锁定并重置口令



选择用户，选择“Unlock User（解除用户的锁定）”，然后单击“Go（执行）”。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 解除用户帐户的锁定并重置口令

在安装和创建数据库的过程中，可以对 Oracle 提供的多个数据库用户帐户解除锁定，并对其进行重置。如果当时没有选择解除用户帐户锁定，可以通过以下方法解除用户锁定：在“Users（用户）”页上选择用户，然后从“Actions（操作）”列表中选择“Unlock User（解除用户的锁定）”并单击“Go（执行）”。这样不会更改口令。如果您解除用户锁定时口令已过期，它将保持过期，直到您编辑该用户并更改口令。

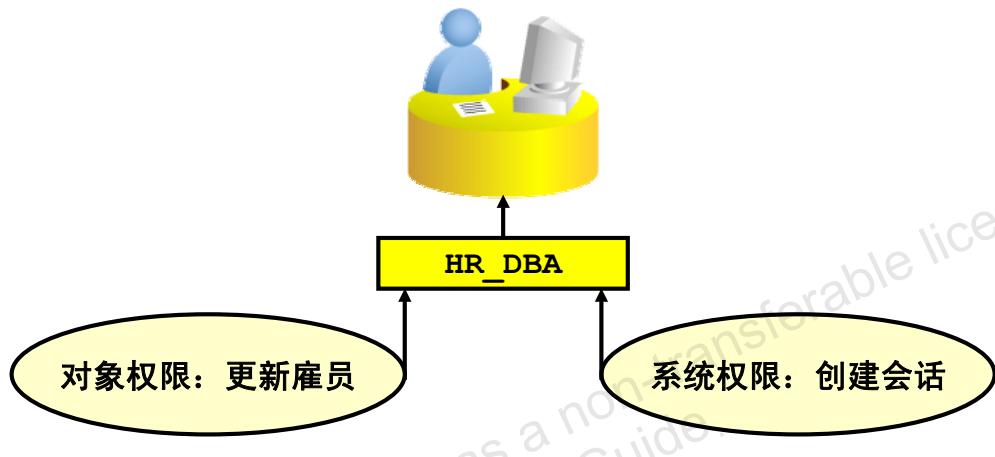
要解除用户锁定并重置口令，请在“Edit Users（编辑用户）”页上执行下列步骤：

1. 在“Enter Password（输入口令）”和“Confirm Password（确认口令）”字段中输入新口令。
2. 选中“Unlocked（解除锁定）”复选框。
3. 单击“Apply（应用）”，重置口令并解除对用户帐户的锁定。

# 权限

用户权限有两类：

- 系统：允许用户在数据库中执行特定的操作
- 对象：允许用户访问和操纵特定的对象



ORACLE

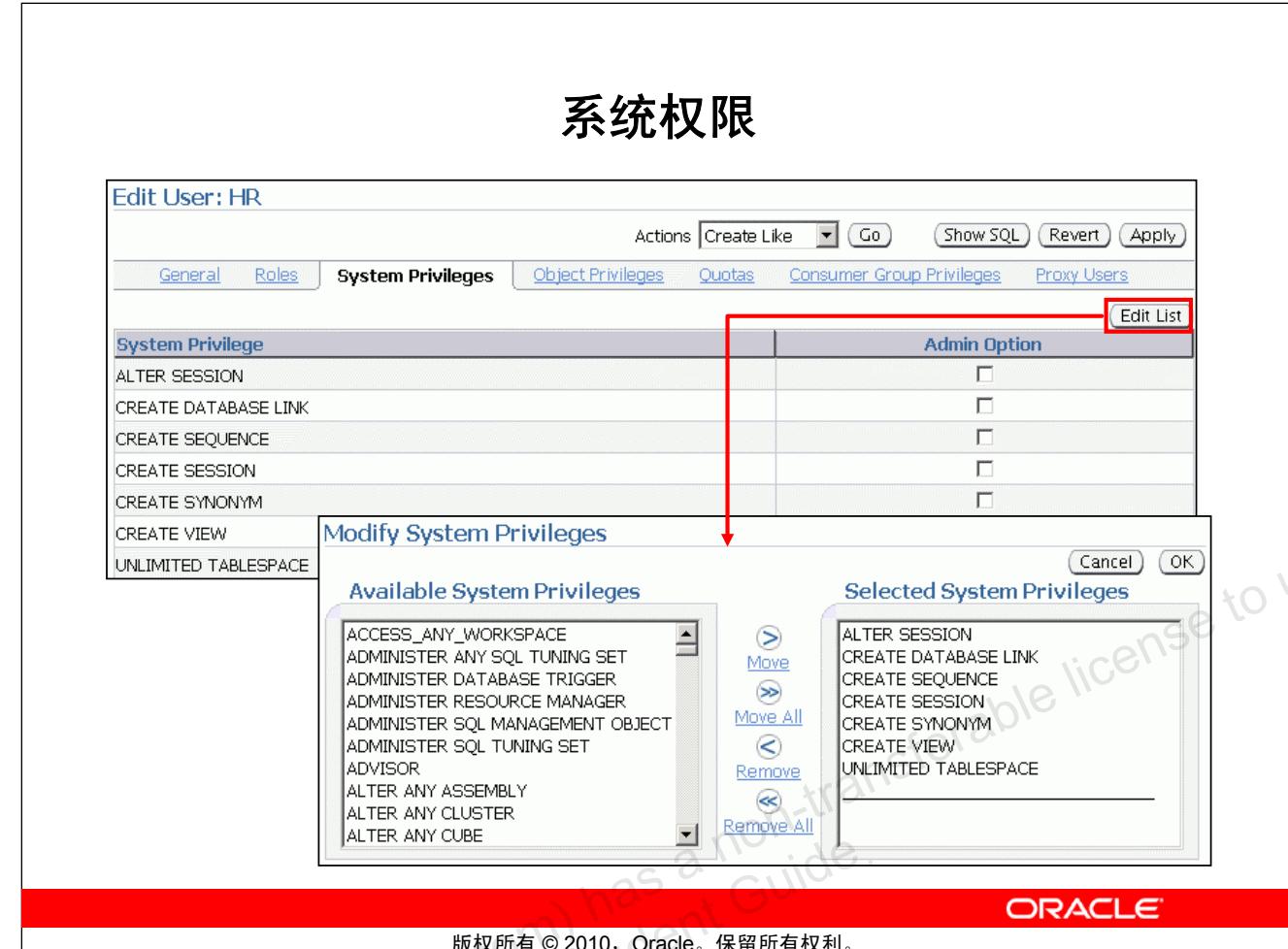
版权所有 © 2010, Oracle。保留所有权利。

## 权限

“权限”是用于执行特定类型的 SQL 语句或访问其他用户的对象的一种权限。Oracle DB 允许您控制用户在数据库中能够（或无法）执行的操作。

权限可分为以下两种类别：

- **系统权限：**每种系统权限都允许用户执行一个特定的数据库操作或一类数据库操作。例如，创建表空间的权限就是一种系统权限。系统权限可由管理员授予，或者由被显式授权管理权限的用户授予。有 170 多种不同的系统权限。很多系统权限都包含 ANY 子句。
- **对象权限：**用户可以使用对象权限对特定对象（如表、视图、序列、过程、函数或程序包）执行特定的操作。在没有特定权限的情况下，用户只能访问他们自己拥有的对象。对象权限可以由对象的所有者或管理员授予，也可以由被显式授予了权限，可以为其他人员分配对某个对象的权限的人员授予。



## 系统权限

要授予系统权限，请单击“Edit User（编辑用户）”页上的“Systems Privileges（系统权限）”选项卡。从可用权限列表中选择适当的权限，然后通过单击“Move（移动）”箭头将其移至“Selected System Privileges（选定的系统权限）”列表中。

使用 ANY 子句授予权限意味着权限可以跨越方案。例如，如果您具有 CREATE TABLE 权限，则可以创建表，但只能在您自己的方案中创建。使用 SELECT ANY TABLE 权限就可从其他用户拥有的表中进行选择。SYS 用户和具有 DBA 角色的用户被授予了所有 ANY 权限，因此，他们可以对任何数据对象执行任何操作。可以使用 Oracle Database Vault 选件来控制 ANY 系统权限的作用域。

如果选中了“Admin Option（管理选项）”复选框，则用户可以管理权限并将系统权限授予给其他用户。

授予系统权限的 SQL 语法是：

```
GRANT <system_privilege> TO <grantee_clause> [WITH ADMIN OPTION]
```

## 系统权限（续）

请在授予系统权限之前仔细考虑安全要求。某些系统权限通常只能授予给管理员：

- **RESTRICTED SESSION:** 使用这个权限您可以登录，即使数据库是在受限模式下打开的也是如此。
- **SYSDBA 和 SYSOPER:** 使用这两个权限可以在数据库中执行关闭、启动、恢复及其它管理任务。用户使用 SYSOPER 可执行基本操作任务，但不能查看用户数据。这个权限包括以下系统权限：

- STARTUP 和 SHUTDOWN
- CREATE SPFILE
- ALTER DATABASE OPEN/MOUNT/BACKUP
- ALTER DATABASE ARCHIVELOG
- ALTER DATABASE RECOVER (仅限完全恢复。任何形式的不完全恢复，如 UNTIL TIME | CHANGE | CANCEL | CONTROLFILE，都需要以 SYSDBA 身份建立连接。)
- RESTRICTED SESSION

除此之外，SYSDBA 系统权限还允许执行不完全恢复和删除数据库。用户使用 SYSDBA 系统权限可以 SYS 用户身份有效地建立连接。

- **SYSASM:** 使用此权限可以启动、关闭和管理 ASM 实例。
- **DROP ANY object:** 用户使用 DROP ANY 权限可删除其他用户拥有的对象。
- **CREATE、MANAGE、DROP 和 ALTER TABLESPACE:** 这些权限允许进行表空间管理，包括创建、删除和更改表空间的属性。
- **CREATE LIBRARY:** Oracle DB 允许开发人员在 PL/SQL 内创建和调用外部代码（例如 C 库）。此库必须由数据库中的 LIBRARY 对象指定。CREATE LIBRARY 权限允许用户创建可从 PL/SQL 执行的任意代码库。
- **CREATE ANY DIRECTORY:** 作为一种安全措施，代码所在的操作系统目录必须链接到一个虚拟 Oracle 目录对象。使用 CREATE ANY DIRECTORY 权限时，有可能会调用不安全的代码对象。  
用户使用 CREATE ANY DIRECTORY 权限可以在 Oracle 软件所有者能够访问的任何目录中创建目录对象（具有读写访问权限）。这意味着用户可以访问这些目录中的外部过程。用户可以尝试直接读写任何数据库文件，如数据文件、重做日志和审计日志。确保您的组织采用了安全策略，以防止此类强大的权限被误用。
- **GRANT ANY OBJECT PRIVILEGE:** 使用此权限可以对其他人拥有的对象授予对象权限。
- **ALTER DATABASE 和 ALTER SYSTEM:** 这些权限的功能很强，可用于修改数据库和 Oracle 实例，例如，重命名数据文件或刷新缓冲区高速缓存。

## 对象权限

**要向对象授予权限，请执行以下操作：**

- 选择对象类型。
- 选择对象。
- 选择权限。

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 对象权限

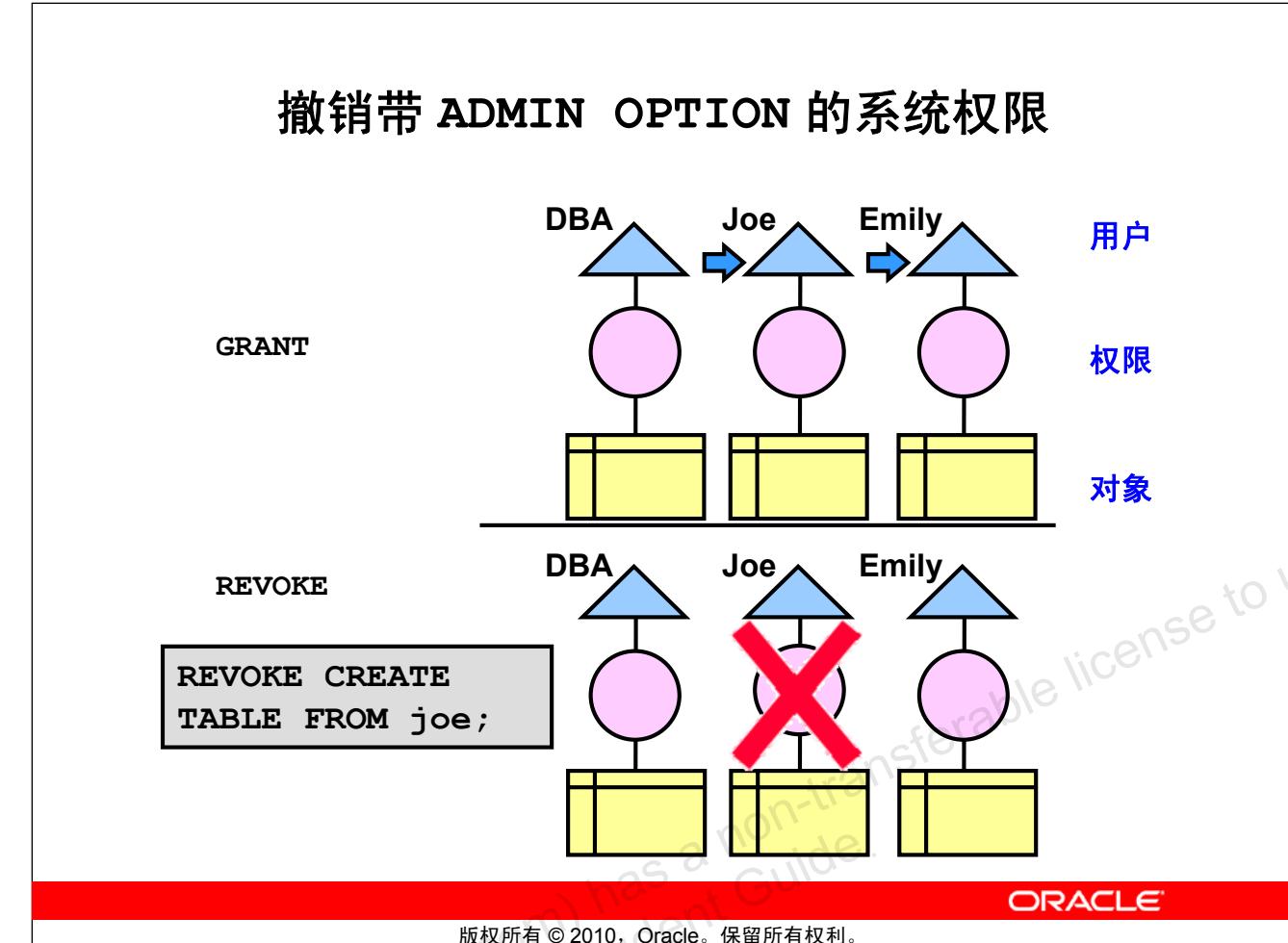
要授予对象权限，请单击“Edit User（编辑用户）”页上的“Object Privileges（对象权限）”选项卡。选择要将其权限授予给用户的对象的类型，然后单击“Add（添加）”按钮。通过输入`<username.object name>`或从列表中进行选择来选择对象。

然后，在“Available Privileges（可用权限）”列表中选择适当权限，再单击“Move（移动）”按钮。选择权限之后，单击“OK（确定）”。

如果允许此用户向其他用户授予相同的访问权限，请在“Edit User（编辑用户）”页中选中“Grant（授权）”复选框。

授予对象权限的 SQL 语法是：

```
GRANT <object_privilege> ON <object> TO <grantee_clause>
[WITH GRANT OPTION]
```



## 撤销带 ADMIN OPTION 的系统权限

可以使用 REVOKE SQL 命令来撤销直接通过 GRANT 命令授予的系统权限。其系统权限带 ADMIN OPTION 的用户可以撤销其他任何数据库用户的权限。撤销者与最初被授予权限的用户不一定是同一个用户。

无论是否指定了 ADMIN OPTION，撤销系统权限时都不会产生级联影响。

撤销系统权限的 SQL 语法是：

```
REVOKE <system_privilege> FROM <grantee clause>
```

本幻灯片说明了以下情况。

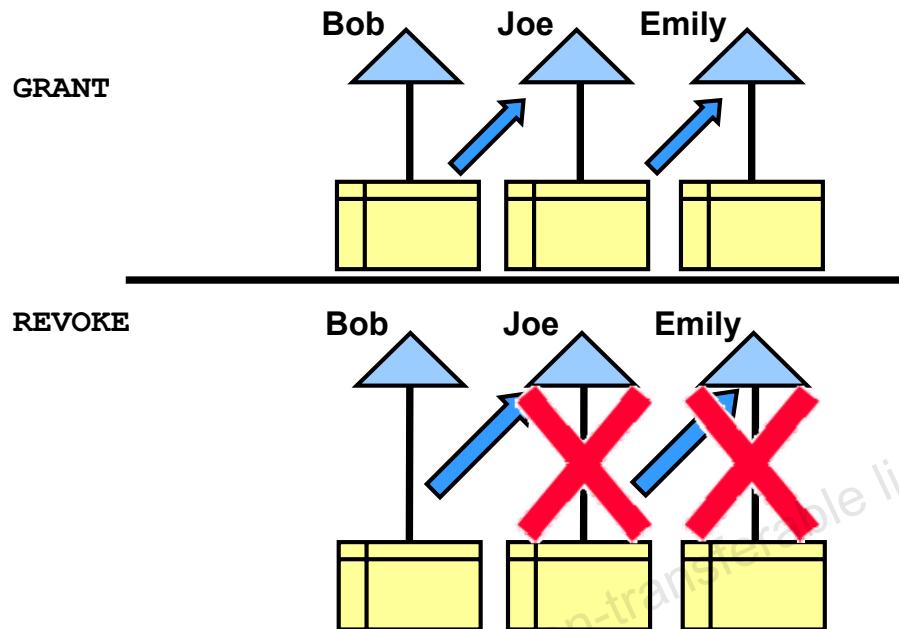
### 场景

1. DBA 将带 ADMIN OPTION 的 CREATE TABLE 系统权限授予 Joe。
2. Joe 创建表。
3. Joe 将 CREATE TABLE 系统权限授予 Emily。
4. Emily 创建表。
5. DBA 撤销 Joe 的 CREATE TABLE 系统权限。

### 结果

Joe 的表仍旧存在，但是 Joe 不能再创建新表。Emily 的表仍旧存在，而且她仍旧具有 CREATE TABLE 系统权限。

## 撤销带 GRANT OPTION 的对象权限



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 撤销带 GRANT OPTION 的对象权限

撤销与数据操纵语言 (DML) 操作相关的系统权限时可能会产生级联影响。例如，如果向某个用户授予了 SELECT ANY TABLE 权限，并且该用户创建了使用该表的过程，则必须重新编译用户的方案中包含的所有过程，然后才能再次使用这些过程。

撤销带 GRANT OPTION 的对象权限时也会级联。用户只能撤销授予他们的那些权限。例如，Bob 无法撤销 Joe 授予 Emily 的对象权限。只有被授予者或者具有名为 GRANT ANY OBJECT PRIVILEGE 权限的用户可以撤销对象权限。

#### 场景

1. Joe 被授予了对 EMPLOYEES 的 SELECT 对象权限（带 GRANT OPTION）。
2. Joe 将对 EMPLOYEES 的 SELECT 权限授予 Emily。
3. 撤销 Joe 的 SELECT 权限。此撤销也级联到 Emily。

## 角色的优点

- 简化权限管理
- 进行动态权限管理
- 有选择地提供权限



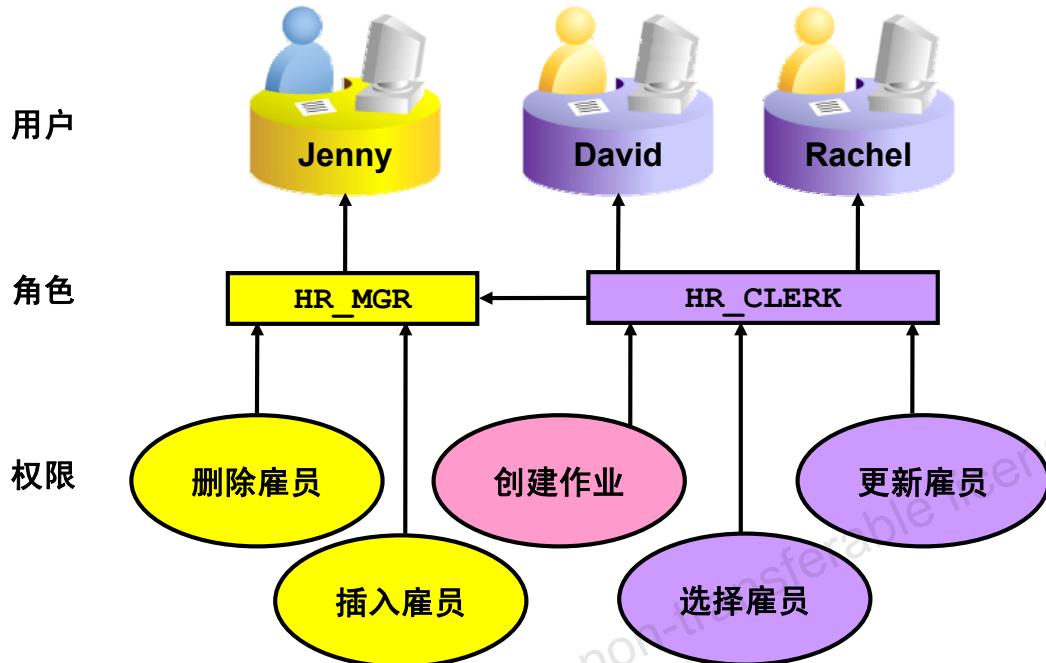
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 角色的优点

- **简化权限管理:** 使用角色可简化权限管理。可以将一组权限授予给某个角色，然后将该角色授予给每个用户，而不是将同一组权限授予给多个用户。
- **进行动态权限管理:** 如果修改了与某个角色关联的权限，则所有被授予该角色的用户都会立即自动获得修改后的权限。
- **有选择地提供权限:** 通过启用或禁用角色可以暂时打开或关闭权限。这样便可以在指定情形下控制用户的权限。

## 将权限分配给角色以及将角色分配给用户



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 将权限分配给角色以及将角色分配给用户

在大多数系统中，将必要的权限分别授予给每个用户是一项很耗时的工作，而且很容易出错。Oracle 软件通过角色提供了简单且受控的权限管理方式。角色是可授予给用户或其它角色的、由相关权限组成的指定组。设计角色是为了简化数据库中的权限管理，从而增强数据库的安全性。

#### 角色特性

- 角色就像用户，可以授予角色权限或撤销角色权限。
- 角色就像系统权限一样，可以将其授予给用户或其它角色，也可以从用户或其它角色撤销。
- 角色可以由系统权限和对象权限组成。
- 可以对授予了某一角色的每个用户启用或禁用该角色。
- 可能需要口令才能启用角色。
- 角色不归任何用户拥有，也不属于任何方案。

在幻灯片示例中，对 employees 表的 SELECT 和 UPDATE 权限以及 CREATE JOB 系统权限被授予 HR\_CLERK 角色。对 employees 表的 DELETE 和 INSERT 权限和 HR\_MGR 角色被授予 HR\_MGR 角色。经理被授予 HR\_MGR 角色，现在他可以选择、删除、插入以及更新 employees 表。

## 预定义角色

角色	包括的权限
CONNECT	CREATE SESSION
RESOURCE	CREATE CLUSTER、CREATE INDEXTYPE、CREATE OPERATOR、CREATE PROCEDURE、CREATE SEQUENCE、CREATE TABLE、CREATE TRIGGER、CREATE TYPE
SCHEDULER_ADMIN	CREATE ANY JOB、CREATE EXTERNAL JOB、CREATE JOB、EXECUTE ANY CLASS、EXECUTE ANY PROGRAM、MANAGE SCHEDULER
DBA	大多数系统权限；几个其它角色。不要授予非管理员
SELECT_CATALOG_ROLE	无系统权限；HS_ADMIN_ROLE 以及对数据字典的 1,700 多个对象权限

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 预定义角色

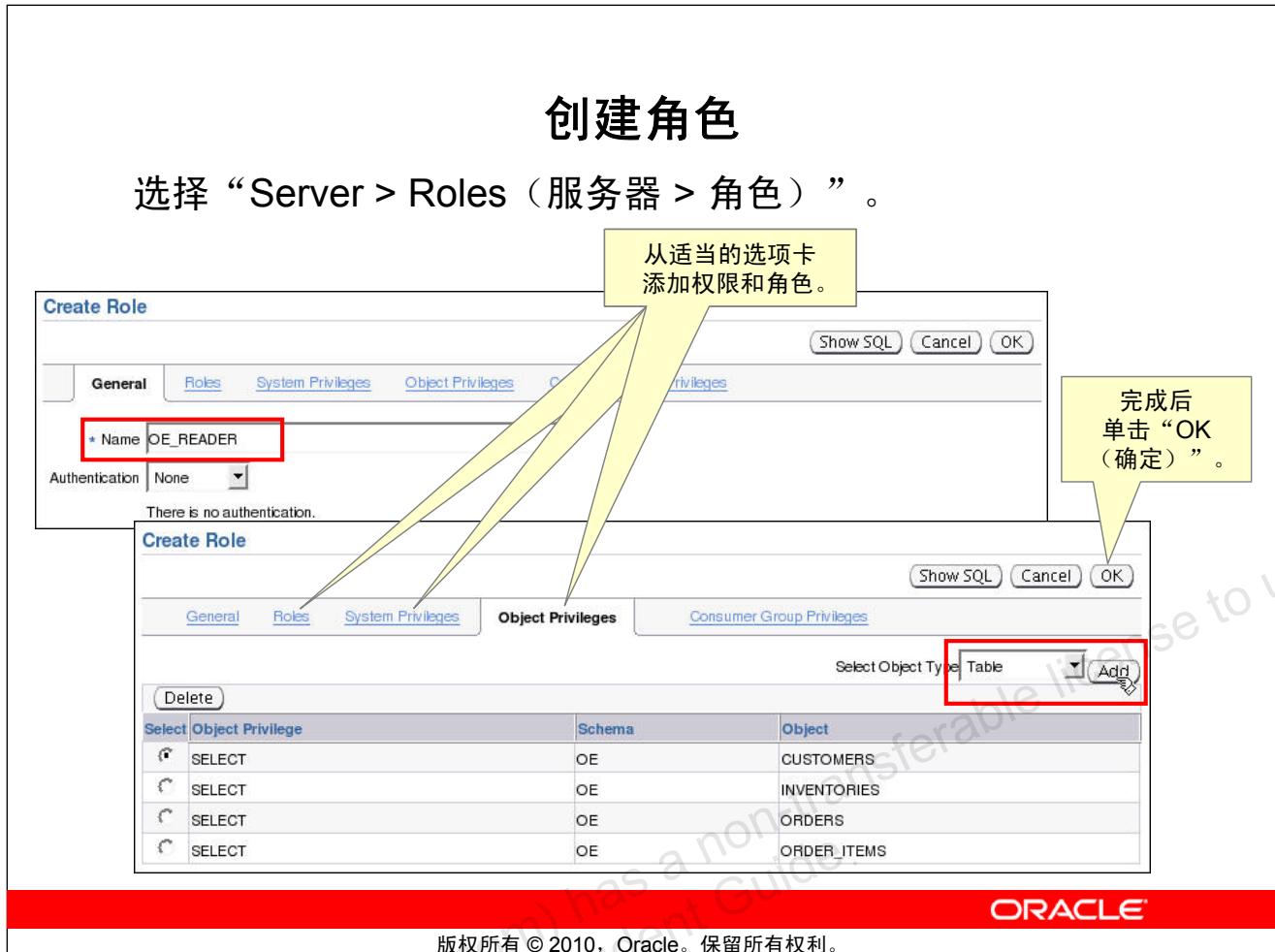
运行数据库创建脚本时会为 Oracle DB 自动定义若干个角色。CONNECT 将自动授予通过 Oracle Enterprise Manager 创建的任何用户。出于安全原因，自 10.2.0 版 Oracle DB 之后，CONNECT 角色仅包含 CREATE SESSION 权限。

**注：**请注意，授予 RESOURCE 角色包括授予 UNLIMITED TABLESPACE 权限。

### 功能性角色

已经创建了一些允许您管理特殊功能（如果已安装了这些功能）的其它角色。例如，XDBADMIN 包含管理扩展标记语言 (XML) 数据库（如果已安装此功能）所需的权限。AQ\_ADMINISTRATOR\_ROLE 提供管理高级队列的权限。HS\_ADMIN\_ROLE 包括管理异构服务所需的权限。

在没有 Oracle 技术支持协助的情况下，不得改变授予给这些功能性角色的权限，否则可能会无意中禁用所需的功能。



## 创建角色

“角色”是一个由相关权限组成的指定组，可授予给用户或其它角色。DBA 通过角色来管理权限。

要创建角色，请执行以下步骤：

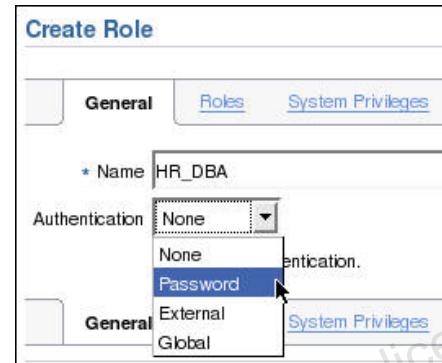
1. 在 Enterprise Manager Database Control 中，单击 “Server（服务器）” 选项卡，然后在 “Security（安全）” 标题下单击 “Roles（角色）”。
2. 单击 “Create（创建）” 按钮。
3. 在 “General（一般信息）” 选项卡上，输入角色的名称。
4. 可以根据需要添加系统权限、对象权限以及其它角色。以后可以根据需要编辑该角色来修改这些设置。
5. 完成后单击 “OK（确定）”。

## 保护角色

- 角色可以是非默认的，并在需要时启用。

```
SET ROLE vacationdba;
```

- 可以通过验证保护角色。



- 还可以通过编程保护角色。

```
CREATE ROLE secure_application_role  
IDENTIFIED USING <security_procedure_name>;
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 保护角色

默认情况下通常会启用角色，这意味着如果将角色授予给某个用户，则该用户就可以行使指定给角色的权限。在连接时会为用户分配默认角色。

此时用户可以：

- 使角色成为非默认角色。将角色授予给用户后，取消选中“DEFAULT（默认）”复选框。现在，用户必须先显式启用角色，才可以行使角色的权限。
- 要求对角色进行附加验证。默认情况下角色的验证为“无”，但可以要求对角色进行附加验证，之后才能设置角色。
- 创建只有成功地执行了 PL/SQL 过程后才能启用的安全应用程序角色。通过 PL/SQL 过程可以检查某些方面，如用户的网络地址、用户正在运行的程序、当日时间以及恰当保护权限组所需的其它元素。
- 使用 Oracle Database Vault 选项可以轻松管理角色。可以简化安全应用程序角色，并进一步地限制普通角色。

## 将角色分配给用户



### 将角色分配给用户

可以使用角色来管理数据库权限。可以向角色添加权限，然后将该角色授予给用户。用户然后可以启用角色，从而行使角色授予的权限。角色包含授予给该角色的所有权限，以及授予给该角色的其它角色的所有权限。

默认情况下，Oracle Enterprise Manager 会自动将 CONNECT 角色授予给新用户。这样，用户可连接到数据库，然后在自己的方案中创建数据库对象。

要将角色分配给用户，请执行以下操作：

1. 在 Enterprise Manager Database Control 中，单击“Server（服务器）”选项卡，然后在“Security（安全）”标题下单击“Users（用户）”。
2. 选择用户，然后单击“Edit（编辑）”按钮。
3. 单击“Roles（角色）”选项卡，然后单击“Edit List（编辑列表）”按钮。
4. 在“Available Roles（可用角色）”中选择所需角色，然后将其移至“Selected Roles（所选角色）”中。
5. 分配了所有适当的角色后，单击“OK（确定）”按钮。

## 小测验

默认情况下，在 Oracle Database 11g 中创建的所有口令都不区分大小写。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小测验

数据库角色：

1. 可以启用或禁用
2. 可以由系统权限和对象权限组成
3. 由其创建者所拥有
4. 无法通过口令来保护

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

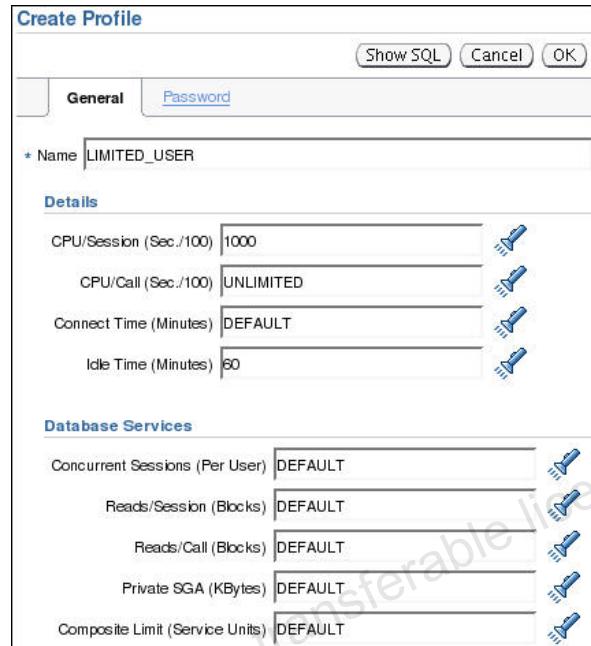
答案：1 和 2

## 概要文件和用户

一次只能向用户分配一个概要文件。

**概要文件：**

- 控制资源使用量
- 管理帐户状态和口令到期



**注： RESOURCE\_LIMIT 必须设置为 TRUE，概要文件才能强制实行资源限制。**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 概要文件和用户

概要文件是用于限制数据库使用和实例资源的一组指定资源限制条件。通过概要文件还可管理帐户状态并对用户的口令进行限制（长度、到期时间等）。每个用户都分配有一个概要文件，而且该用户在指定时间只属于一个概要文件。如果在更改用户概要文件时用户已经登录，则所做更改在用户下一次登录时才生效。

DEFAULT 概要文件用作其它所有概要文件的基础。如本幻灯片中所示，可隐式指定概要文件的限制（如“CPU/Session (CPU/会话) ”），可取消限制（如“CPU/Call (CPU/调用) ”），也可以引用 DEFAULT 概要文件中的任何设置（如“Connect Time (连接时间) ”）。

只有当 RESOURCE\_LIMIT 初始化参数设置为 TRUE，概要文件才能对用户强制实行资源限制。如果 RESOURCE\_LIMIT 使用默认值 FALSE，则忽略概要文件的资源限制。概要文件的口令设置始终会强制实行。

管理员使用概要文件可控制以下系统资源：

- **CPU：** 可按会话或调用限制 CPU 资源。将“CPU/Session (CPU/会话) ”限制为 1,000 表示，如果使用此概要文件的任一会话占用 10 秒以上的 CPU 时间（CPU 时间限制以百分之一秒为单位），该会话就会收到错误消息并被注销：

ORA-02392: exceeded session limit on CPU usage, you are being logged off

## 概要文件和用户（续）

对每个调用所做的限制也起相同作用，但是它不是限制用户的整个会话，而是防止任一命令占用过多的 CPU。如果“CPU/Call (CPU/调用)”受到限制，并且用户超出了该限制，则命令会中止。用户将收到如下所示的错误消息：

ORA-02393: exceeded call limit on CPU usage

- **网络/内存：**每个数据库会话都会占用系统内存资源和网络资源（如果会话不是来自服务器的本地用户）。可以指定以下参数：
  - **连接时间：**指示用户在自动注销前可以保持连接的分钟数。
  - **空闲时间：**指示用户会话在自动注销前可以保持空闲的分钟数。只会计算服务器进程的空闲时间。空闲时间不考虑应用程序活动。`IDLE_TIME` 限制不受长时间运行的查询和其它操作的影响。
  - **并行会话：**指示使用数据库用户帐户可以创建多少并行会话。
  - **专用 SGA：**限制在系统全局区域 (SGA) 中执行排序、合并位图等操作所占用的空间量。此限制仅在会话使用共享服务器时才有效（在“配置 Oracle Network 环境”一课中介绍了共享服务器）。
- **磁盘 I/O：**限制用户在每个会话级或每个调用级可读取的数据量。“读取/会话”和“读取/调用”可限制内存和磁盘的总读取次数。这样做可确保执行大量 I/O 操作的语句不会过度使用内存和磁盘。

通过概要文件还可提供组合限制。组合限制以“CPU/会话”、“读取/会话”、“连接时间”和“专用 SGA”的加权组合为基础。《Oracle 数据库安全性指南》中对组合限制进行了详细介绍。

要创建概要文件，请单击“Server (服务器)”选项卡，然后在“Security (安全)”标题下单击“Profiles (概要文件)”。在“Profiles (概要文件)”页中，单击“Create (创建)”按钮。

**注：**资源管理器可替代许多概要文件设置。有关资源管理器的详细信息，请参阅《Oracle 数据库管理员指南》。

## 实施口令安全功能



**注：请不要使用会导致 SYS、SYSMAN 和 DBSNMP 口令到期以及相应帐户被锁定的概要文件。**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 实施口令安全功能

Oracle 口令管理是通过用户概要文件来实现的。概要文件可提供许多标准安全功能。

**帐户锁定：**如果用户在指定的次数内尝试登录系统失败，系统会在设置的持续时间内自动锁定帐户。

- **FAILED\_LOGIN\_ATTEMPTS:** 指定在锁定帐户前尝试登录的失败次数
- **PASSWORD\_LOCK\_TIME:** 指定尝试登录失败达到了指定的次数后锁定帐户的天数

**口令失效和到期：**使用户口令具有生存期，口令在此生存期后会到期，必须对其进行更改。

- **PASSWORD\_EXPIRE\_DATE:** 确定口令生存期（天），之后该口令就会到期
- **PASSWORD\_GRACE\_TIME:** 指定首次成功登录后更改口令的宽限期（天），之后该口令就会到期

**注：**使 SYS、SYSMAN 和 DBSNMP 帐户口令到期和锁定这些帐户，会导致 Oracle Enterprise Manager 无法正常运行。应用程序必须捕获“口令到期”警告消息并处理口令更改；否则，宽限期一到期，就会锁定用户而不通知原因。

## 实施口令安全功能（续）

**口令历史记录：**通过核对新口令可确保在指定的时间内或者在指定的口令更改次数内不重复使用口令。通过使用下列其中一项来进行核对：

- **PASSWORD\_REUSE\_TIME:** 指定用户不能在指定天数内重复使用口令
- **PASSWORD\_REUSE\_MAX:** 指定在可重复使用当前口令之前口令更改需达到的次数

请回想一下，概要文件参数的值是在 DEFAULT 概要文件中设置的或从该概要文件中继承的。

如果两个口令历史记录参数都具有值 UNLIMITED，则 Oracle DB 将同时忽略两者。用户可以随时重用任何口令，但这不是一种有效的安全做法。

如果两个参数均已设置，则允许重用口令，但必须同时满足这两个条件。用户更改口令的次数必须达到指定次数，并且在最后一次使用旧口令之后已经过了指定的天数。

例如，假设用户 ALFRED 的概要文件中的 PASSWORD\_REUSE\_MAX 设置为 10，PASSWORD\_REUSE\_TIME 设置为 30，则用户 ALFRED 只有在已重置了 10 次口令，并且自最后一次使用该口令以来已经过了 30 天之后，才能重用该口令。

如果一个参数设置为一个数，而另一个参数指定为 UNLIMITED，则用户永远不能重用口令。

**口令复杂性验证：**通过对口令进行复杂性检查可验证口令是否符合特定的规则。这种检查一定要确保口令足够复杂，才能防止入侵者通过猜测口令尝试闯入系统。

PASSWORD\_VERIFY\_FUNCTION 参数指定一个 PL/SQL 函数，以便在分配口令之前执行口令复杂性检查。口令验证函数必须由 SYS 用户拥有，而且必须返回布尔值（TRUE 或 FALSE）。位于以下目录中的 utlpwdmg.sql 脚本提供了模型口令验证函数：

- Unix 和 Linux 平台: \$ORACLE\_HOME/rdbms/admin
- Windows 平台: %ORACLE\_HOME%\rdbms\admin

# 创建口令概要文件

**Create Profile**

<input checked="" type="radio"/> General	<input type="radio"/> Password
<b>Password</b>	
Expire in (days)	<input type="text" value="90"/>
Lock (days past expiration)	<input type="text" value="10"/>
<b>History</b>	
Number of passwords to keep	<input type="text" value="2"/>
Number of days to keep for	<input type="text" value="UNLIMITED"/>
<b>Complexity</b>	
Complexity function	<input type="text" value="VERIFY_FUNCTION_11G"/>
<b>Failed Login</b>	
Number of failed login attempts to lock after	<input type="text" value="3"/>
Number of days to lock for	<input type="text" value="5/1440"/>

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 创建口令概要文件

要创建口令概要文件，请单击“Server（服务器）”选项卡，然后在“Security（安全）”标题下单击“Profiles（概要文件）”。在“Profiles（概要文件）”页中，单击“Create（创建）”按钮。单击“Password（口令）”选项卡以设置口令限制。

可以从值列表中选择各个设置的常用值（单击手电筒图标可进行浏览），也可输入定制值。所有时段都用“天”表示，但也可表示为分数。一天中有 1,440 分钟，因此 5/1,440 为 5 分钟。

使用 Enterprise Manager 也可编辑现有的口令概要文件。

如果 utlpwdmg.sql 脚本已运行，则 VERIFY\_FUNCTION 和 VERIFY\_FUNCTION\_11G 函数将是可用的。如果您创建了自己的复杂性函数，则可以输入该函数的名称。该函数名称不会显示在“Select（选择）”列表中。如果该函数生成运行时错误，用户将无法更改口令。

## 删除口令概要文件

在 Enterprise Manager 中，无法删除用户使用的概要文件。但是，如果删除概要文件时使用了 CASCADE 选项（例如，在 SQL\*Plus 中），则具有该概要文件的所有用户将自动分配 DEFAULT 概要文件。

## 提供的口令验证函数: VERIFY\_FUNCTION\_11G

VERIFY\_FUNCTION\_11G 函数可确保口令满足以下条件:

- 至少包含八个字符
- 与用户名、带有一个数字的用户名以及逆序的用户名不同
- 与数据库名以及带有一个数字的数据库名不同
- 是至少含有一个字母和一个数字的字符串
- 与之前的口令至少有三个字母不相同

**提示:** 使用此函数作为模板可创建自己的定制口令验证。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 提供的口令验证函数: VERIFY\_FUNCTION\_11G

Oracle 服务器提供两个名称分别为 VERIFY\_FUNCTION 和 VERIFY\_FUNCTION\_11G 的口令复杂性验证函数。这些函数是使用 <oracle\_home>/rdbms/admin/utlpwdmg.sql 脚本创建的。VERIFY\_FUNCTION 是为偏好以前版本提供的口令函数的用户提供。必须在 SYS 方案中创建口令复杂性验证函数。此函数可用作定制口令验证的模板。

除了创建 VERIFY\_FUNCTION, utlpwdmg 脚本还通过下面的 ALTER PROFILE 命令更改 DEFAULT 概要文件:

```
ALTER PROFILE default LIMIT
  PASSWORD_LIFE_TIME 180
  PASSWORD_GRACE_TIME 7
  PASSWORD_REUSE_TIME UNLIMITED
  PASSWORD_REUSE_MAX UNLIMITED
  FAILED_LOGIN_ATTEMPTS 10
  PASSWORD_LOCK_TIME 1
  PASSWORD_VERIFY_FUNCTION verify_function_11g;
```

请注意, 创建用户时, 除非指定了其它概要文件, 否则会向用户分配 DEFAULT 概要文件。

## 将限额分配给用户

没有 UNLIMITED TABLESPACE 系统权限的用户必须先获得限额，然后才能在表空间中创建对象。

限额可以是：

- 以兆字节或千字节为单位的特定值
- 无限制

Tablespace	Quota	Value	Unit
EXAMPLE	Value	20	MBytes
INVENTORY	None	0	MBytes
SYSAUX	None	0	MBytes
SYSTEM	None	0	MBytes
TEMP	None	0	MBytes
UNDOTBS1	None	0	MBytes
USERS (Default)	Unlimited	0	MBytes

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 将限额分配给用户

“限额”是允许给定表空间具有的空间。默认情况下，对于任何表空间用户都没有限额。使用以下三个选项可为用户提供表空间限额。

- **无限制：**允许用户最大限度地使用表空间中的可用空间。
- **值：** 用户可以使用的空间（以千字节或兆字节为单位）。这并不保证一定会为用户保留该空间。因为，此值可能大于或小于表空间中的当前可用空间。
- **UNLIMITED TABLESPACE 系统权限：** 覆盖所有单独的表空间限额，对于所有表空间（包括 SYSTEM 和 SYSAUX），为用户提供无限制的限额。授予此权限时必须谨慎。  
注：请注意，授予 RESOURCE 角色包括授予此权限。

对于 SYSTEM 或 SYSAUX 表空间，不得为用户提供限额。通常，仅 SYS 和 SYSTEM 用户能够在 SYSTEM 或 SYSAUX 表空间中创建对象。

对于分配的临时表空间或任何还原表空间，您不需要具有限额。您不必具有限额就可以在 Oracle DB 中插入、更新和删除数据。唯一需要限额的用户是拥有数据库对象的帐户。通常安装应用程序代码时，安装程序会创建数据库帐户来拥有对象。仅这些帐户需要限额。其他数据库用户可以被授予权限来使用这些对象而不需要限额。

### 将限额分配给用户（续）

- 当用户创建或扩展段时，Oracle 实例会检查限额。
- 对于分配给用户方案的活动，只有那些使用表空间中的空间的活动才会计入限额。那些不使用所分配的表空间中的空间的活动（如创建视图或使用临时表空间）不会影响限额。
- 在使用 PURGE 子句删除了用户拥有的对象或者在回收站中自动清除了用户所拥有的对象时，会补充限额。

## 应用最少权限原则

- 保护数据字典:

**O7\_DICTIONARY\_ACCESSIBILITY=FALSE**

- 撤销不必要的 PUBLIC 权限。
- 使用访问控制列表 (ACL) 来控制网络访问。
- 限制用户可访问的目录。
- 限制具有管理权限的用户。
- 限制远程数据库验证:

**REMOTE\_OS\_AUTHENT=FALSE**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 应用最少权限原则

应用最少权限原则意味着必须仅为用户授予高效地完成任务所需的权限。这会降低用户修改或查看（无意或恶意）他们无权修改或查看的数据的机率。

**保护数据字典:** 默认情况下，O7\_DICTIONARY\_ACCESSIBILITY 参数设置为 FALSE。如果没有很好的理由，一定不要更改这个设置，因为这会阻止具有 ANY\_TABLE 系统权限的用户访问数据字典的基表。这个参数设置还确保了 SYS 用户只能以 SYSDBA 身份登录。

**撤销不必要的 PUBLIC 权限:** 有多个程序包对于需要这些程序包的应用程序非常有用，但需要正确的配置才能安全地使用这些程序包。PUBLIC 被授予了对以下程序包的执行权限: UTL\_SMTP、UTL\_TCP、UTL\_HTTP 和 UTL\_FILE。在 Oracle Database 11g 中，网络访问由访问控制列表 (ACL) 控制，ACL 可配置为允许某些用户访问特定的网络服务。默认情况下，拒绝访问网络。要允许访问网络，必须创建 ACL。可在两个级别控制通过 UTL\_FILE 对文件的访问：在操作系统级别使用文件和目录的权限，以及在数据库中通过 DIRECTORY 对象来允许访问特定文件系统目录。可以向用户授予对 DIRECTORY 对象的读取权限或读写权限。应谨慎控制对其它 PL/SQL 程序包的执行权限。

## 应用最少权限原则（续）

有一些功能较强大的程序包可能被误用，这些程序包包括：

- **UTL\_SMTP:** 允许将数据库用作简单邮件传输协议 (SMTP) 邮件服务器来发送任意电子邮件。使用 ACL 控制哪些用户可以访问哪些计算机。
- **UTL\_TCP:** 允许数据库服务器与任何正在接收或等待的网络服务建立外出网络连接。因此，在数据库服务器与任何正在等待的网络服务之间会发送任意数据。使用 ACL 可控制访问权限。
- **UTL\_HTTP:** 允许数据库服务器通过 HTTP 请求并检索数据。如果将这个程序包授予用户，可能会允许以 HTML 形式将数据发送到恶意 Web 站点。可使用 ACL 限制访问权限。
- **UTL\_FILE:** 如果配置不当，则可能允许对主机操作系统上的任何文本文件进行访问。正确配置时，此程序包可限制用户只能访问特定目录位置。

**限制对操作系统目录的访问：**通过数据库中的 DIRECTORY 对象，DBA 可以将目录映射到操作系统路径，从而将这些目录的权限授予各个用户。

**限制具有管理权限的用户：**请不要为数据库用户提供不必要的权限。一定不能将 DBA 角色授予非管理员。要实施最少权限，请限制授予下列类型的权限：

- 授予系统和对象的权限
- 连接到数据库的 SYS 权限，如 SYSDBA 和 SYSOPER
- 其它 DBA 类型的权限，如 DROP ANY TABLE

**限制远程数据库验证：**默认情况下，REMOTE\_OS\_AUTHENT 参数设置为 FALSE。一定不要更改这个设置，除非可确保所有客户机都能适当地验证用户。在引入了安全外部口令存储（在 Oracle Database 10g 发行版 2 中提供）后，允许远程操作系统验证就有了令人信服的理由。

在远程验证过程中：

- 数据库用户通过外部方式来进行验证
- 远程系统验证用户
- 用户登录数据库，不必接受另外的验证

**注：**如果您撤销了一些权限，务必全面测试您的应用程序。

## 保护授权帐户

授权帐户的保护方式有：

- 使用口令区分大小写的口令文件
- 对管理员角色启用严格的验证方法



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 设置数据库管理员验证

任何时候都必须对具有 SYSDBA、SYSOPER 或 SYSASM 权限的用户进行验证。在本地连接时，通过本地操作系统来验证用户是否为授权操作系统组的成员。如果是远程连接，则使用口令文件来验证授权用户。如果已配置口令文件，则首先检查此文件。在 Oracle Database 11g 中，这些口令是区分大小写的。Oracle Database 11g 提供了一些其它方法，使远程管理员验证更加安全并可以集中管理这些授权用户。

使用 Database Configuration Assistant 创建数据库时，口令文件是区分大小写的。如果从早期数据库版本进行升级，请确保用于远程连接的口令文件区分大小写：

```
orapwd file=orapworcl entries=5 ignorecase=N
```

如果您担心口令文件可能会受到攻击或者维护许多口令文件会非常困难，则可以实施严格验证。如果要使用严格的验证方法，则需要高级安全选件。有关严格验证的详细信息，请参阅《Oracle 数据库高级安全管理员指南》。

## 小测验

应用最少权限原则不足以保护 Oracle DB。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 1

## 小测验

如果 RESOURCE\_LIMIT 设置为它的默认值 FALSE，将忽略概要文件的口令限制。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2

## 小结

在本课中，您应该已经学会：

- 创建和管理数据库用户帐户：
  - 验证用户
  - 分配默认存储区（表空间）
- 授予和撤销权限
- 创建和管理角色
- 创建和管理概要文件：
  - 实施标准口令安全功能
  - 控制用户的资源使用量

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 8 概览：管理用户

本练习包含以下主题：

- 创建限制资源使用量的概要文件
- 创建两个角色：
  - HRCLERK
  - HRMANAGER
- 创建四个新用户：
  - 一个经理和两个职员
  - 在下一个练习中出现的一个方案用户

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



## 管理数据并发处理

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 描述锁定机制以及 Oracle 如何管理数据并发处理
- 监视和解决锁定冲突

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

# 锁

- 可防止多个会话同时更改同一数据
- 是在指定语句的最低可能级别自动获取的
- 不会升级



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 锁

会话必须先锁定要修改的数据，之后数据库才允许会话修改相应数据。锁定后，会话拥有对数据的独占控制权，这样在释放锁之前，其它任何事务处理都不能修改锁定的数据。

事务处理可以锁定单个数据行、多个数据行、甚至整个表。Oracle DB 支持手动锁定和自动锁定。自动获取的锁总是选择尽可能低的锁定级别，以尽量减少与其它事务处理的潜在冲突。

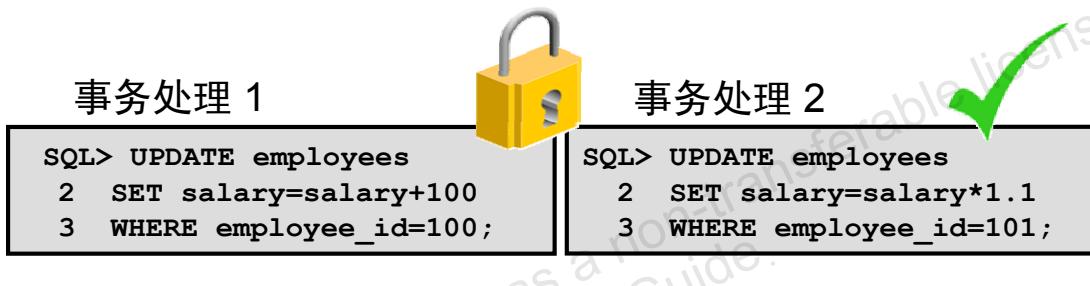
注：Oracle 实例使用许多类型的锁来保持内部一致性。本课只重点介绍用于保护行和表的锁定机制。

## 锁定机制

- 高级数据并发处理:
  - 执行插入、更新和删除时使用行级锁
  - 查询不需要任何锁
- 自动队列管理
- 在事务处理结束（使用 COMMIT 或 ROLLBACK 操作）之前会一直保持锁定

### 示例

假设 employee\_id 100 和 101 所在的行位于同一块中：



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 锁定机制

锁定机制用于在数据库中提供尽可能高的数据并发处理能力。事务处理修改数据时会获取行级锁，而不是块级或表级锁。修改对象（如表移动）时会获取对象级锁，而不是整个数据库锁或方案锁。

数据查询不需要锁，即使有人锁定了数据，查询也能成功进行（总是显示原始的、根据还原信息重新构造的锁定之前的值）。

如果多个事务处理需要锁定同一资源，则第一个请求锁的事务处理会获得锁。其它事务处理将等待，直到第一个事务处理完成为止。排队机制是自动进行的，不需要管理员干预。

事务处理完成（即发出 COMMIT 或 ROLLBACK）时，将释放所有锁。如果事务处理失败，同一后台进程会自动回退失败的事务处理所进行的所有更改，然后释放失败事务处理持有的所有锁。

## 数据并发处理

时间: 09:00:00	事务处理 1	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=100;
	事务处理 2	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=101;
	事务处理 3	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=102;
	...	...
	事务处理 x	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=xxx;

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据并发处理

默认情况下，锁定机制采用细粒度行级锁定模式。不同的事务处理可更新同一表内不同的行，彼此互不干扰。

尽管默认模式是行级锁定，但 Oracle DB 也允许根据需要在更高级别执行手动锁定：

```
SQL> LOCK TABLE employees IN EXCLUSIVE MODE;
Table(s) Locked.
```

使用以上语句时，其他任何尝试更新锁定表中行的事务处理都必须等待，直到发出锁定请求的事务处理完成为止。EXCLUSIVE 是最严格的锁模式。下面列出了其它一些锁模式：

- **ROW SHARE:** 允许对锁定的表进行并发访问，但禁止在会话中锁定整个表进行独占访问。
- **ROW EXCLUSIVE:** 与 ROW SHARE 相同，但是同时禁止以 SHARE 模式锁定。更新、插入或删除数据时会自动获取 ROW EXCLUSIVE 锁。ROW EXCLUSIVE 锁允许多个进程执行读取，但只允许一个进程执行写入。
- **SHARE:** 允许并发查询，但禁止更新锁定的表。需要有 SHARE 锁才能创建表的索引，创建时会自动请求该锁。但是，创建联机索引的操作在建立索引时需要有 ROW SHARE 锁。

共享锁允许多个进程进行读取，但不允许执行写入。删除或更新某个父表中的行，并且其子表在该父表上具有外键约束条件时，也会以透明方式使用共享锁。

## 数据并发处理（续）

- **SHARE ROW EXCLUSIVE:** 用于查询整个表，允许其他人查询表中的行，但禁止其他人在 SHARE 模式下锁定表或更新行。
- **EXCLUSIVE:** 允许查询锁定表，禁止对锁定表执行任何其它活动。需要有 EXCLUSIVE 锁才能删除表。

与任何锁定请求一样，手动锁定语句会一直等待，直到已经持有锁（或先前请求锁定）的所有会话释放锁为止。LOCK 命令可接受用于控制等待行为的特殊参数 NOWAIT。

NOWAIT 会立即将控制权交给您，即使指定的表已经被另一会话锁定：

```
SQL> LOCK TABLE hr.employees IN SHARE MODE NOWAIT;
LOCK TABLE hr.employees IN SHARE MODE NOWAIT
*
ERROR at line 1:
ORA-00054: resource busy and acquire with NOWAIT specified
```

通常不必手动锁定对象。自动锁定机制提供了大多数应用程序所需的数据并发处理能力。Oracle 建议尽量不要使用手动锁定，特别是在开发应用程序时。使用不必要的高锁定级别时，经常会出现严重的性能问题。

## DML 锁

### 事务处理 1

```
SQL> UPDATE employees
  2  SET salary=salary*1.1
  3  WHERE employee_id= 107;
1 row updated.
```

### 事务处理 2

```
SQL> UPDATE employees
  2  SET salary=salary*1.1
  3  WHERE employee_id= 106;
1 row updated.
```

每个 DML 事务处理必须获得两个锁：

- 针对正在更新的一行或多行的 EXCLUSIVE 行锁
- 针对包含这些行的表的 ROW EXCLUSIVE (RX) 模式下的表锁 (TM)



版权所有 © 2010, Oracle。保留所有权利。

### DML 锁

每个 DML 事务处理必须获取两个锁：

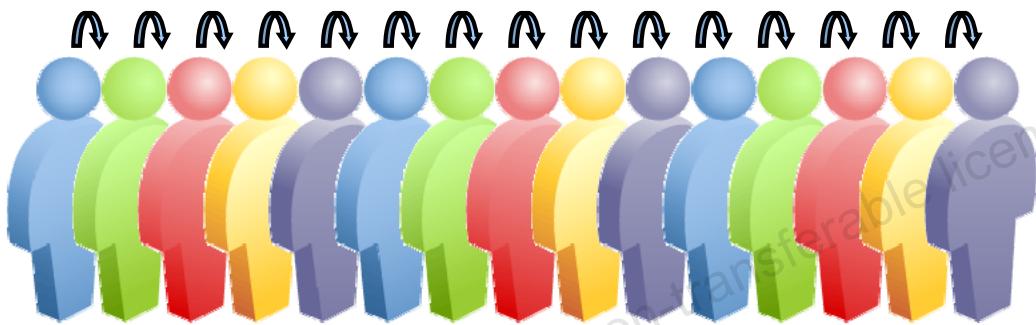
- 针对正在更新的一行或多行的 EXCLUSIVE 行锁。
- 针对正在更新的表的 ROW EXCLUSIVE (RX) 模式下的表锁 (TM)。这可避免在进行更改时另一会话锁定整个表（可能会删除或截断表）。这种模式也称为子排它表锁 (SX)。

对表执行 ROW EXCLUSIVE 锁定时，会禁止 DDL 命令在未提交的事务处理进行到一半时更改字典元数据。这样便可在事务处理的有效期内保持字典完整性和读取一致性。

## 入队机制

入队机制用于跟踪：

- 等待锁的会话
- 请求的锁模式
- 会话请求锁的顺序



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 入队机制

锁定请求自动排队。只要持有某个锁的事务处理一完成，队列中的下一个会话就接收该锁。

入队机制会跟踪请求锁的顺序及请求的锁模式。

已经持有锁的会话可请求转换锁，而不必排到队尾。例如，假定某个会话对表持有 SHARE 锁。该会话可以请求将 SHARE 锁转换为 EXCLUSIVE 锁。如果没有其它事务处理已经对表持有 EXCLUSIVE 或 SHARE 锁，则持有 SHARE 锁的会话就会被授予 EXCLUSIVE 锁，而不必重新在队列中等待。

**注：**等待入队的进程分为两类：没有共享所有权的等待进程，以及有共享所有权、但没有选择升级锁级别的等待进程。第二类等待进程称为转换进程，这类进程的优先级始终高于正常等待进程，即使其等待时间较短。

## 锁冲突

### 事务处理 1

### 时间

### 事务处理 2

UPDATE employees SET salary=salary+100 WHERE employee_id=100; 1 row updated.	9:00:00	UPDATE employees SET salary=salary+100 WHERE employee_id=101; 1 row updated.
UPDATE employees SET COMMISION_PCT=2 WHERE employee_id=101; 会话因锁冲突而等待入队。	9:00:05 	SELECT sum(salary) FROM employees; SUM(SALARY) ----- 692634
会话仍在等待！	16:30:00	在最后 7.5 个小时里进行了许多选择、插入、更新和删除操作，但未提交，也未回退！
1 row updated. 会话继续进行。	16:30:01	commit;

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 锁冲突

锁冲突经常发生，但通常会随着时间流逝通过入队机制得到解决。只有极少数情况下，锁冲突可能需要管理员干预。如幻灯片所示，事务处理 2 在 9:00:00 获取了对某一行的锁且忘记了提交，从而留下了锁。事务处理 1 在 9:00:05 尝试更新整个表，因此需要锁定所有行。但事务处理 2 会阻塞事务处理 1，直到 16:30:01 事务处理 2 提交为止。

这种情况下，用户要尝试执行事务处理 1，就一定要与管理员联系以获得帮助，DBA 必须检测冲突并解决冲突。

## 锁冲突的可能原因

- 未提交更改
- 长时间运行事务处理
- 不必要的高锁定级别



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 锁冲突的可能原因

锁冲突的最常见原因是未提交更改，但还存在其它一些可能原因：

- **长时间运行事务处理：**许多应用程序使用批处理来执行批量更新。这些批作业通常会安排在没有用户活动或者用户活动少时执行，但是，有些情况下，批作业在用户活动少的期间内没有完成或要占用过长的时间来运行。同时执行事务处理和批处理时通常会发生锁冲突。
- **不必要的高锁定级别：**并不是所有数据库都支持行级锁定（Oracle 在 1988 年的发行版 6 中添加了对行级锁定的支持）。某些数据库仍然在页级或表级上进行锁定。开发人员在编写要在许多不同数据库上运行的应用程序时，会人为地使用高锁定级别，以便使 Oracle DB 与功能较少的数据库系统的操作方式相同。如果开发人员不熟悉 Oracle，有时也会以高于 Oracle DB 要求的锁定级别编写代码，其实这是不必要的。

## 检测锁冲突

在“Performance（性能）”页上选择“Blocking Sessions（阻塞会话）”。

Blocking Sessions											
Page Refreshed Aug 18, 2008 11:04:23 PM MDT Refresh											
<input type="button" value="View Session"/> <input type="button" value="Kill Session"/>											
<a href="#">Expand All</a>   <a href="#">Collapse All</a>											
Select Username		Sessions Blocked	Session ID	Serial Number	SQL ID	Wait Class	Wait Event	P1 Value	P2 Value	P3 Value	Seconds in Wait
▼ Blocking Sessions											
BERNST		1	<a href="#">114</a>	33091		Idle	<a href="#">SQL*Net message from client</a>	16508152321	0		89
SMAVRIS		0	<a href="#">124</a>	46897	<a href="#">0tqktcvhr5fcf</a>	Application	<a href="#">eng: TX - row lock contention</a>	1415053318	65545	3085	69

单击“Session ID（会话 ID）”链接，查看关于锁定会话的信息，包括实际 SQL 语句。

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 检测锁冲突

使用 Enterprise Manager 中的“Blocking Sessions（阻塞会话）”页可找出锁冲突。有冲突的锁定请求以分层布局的形式显示，其中持有锁的会话位于顶部，下面是排队请求锁的所有会话。

对于冲突中涉及的每个会话，会显示用户名、会话 ID 和会话已等待的秒数。选择会话 ID 可查看会话当前正在执行或请求的实际 SQL 语句。

自动数据库诊断监视器 (ADDM) 还会自动检测锁冲突，并且会就低效的锁定趋势提出建议。

## 解决锁冲突

为了解决锁冲突，您应该：

- 提交或回退持有锁的会话
- 终止持有锁的会话（在紧急情况下）



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 解决锁冲突

要解决锁冲突，持有锁的会话必须释放锁。让会话释放锁的最好方式是与用户联系，要求用户完成事务处理。

紧急情况下，管理员可以通过单击“Kill Session（终止会话）”按钮来终止持有锁的会话。请记住，终止会话后，当前事务处理中的所有工作都会丢失（回退）。会话被终止的用户必须再次登录，然后重做被终止的会话自上次提交以来所做的所有工作。

如果用户的会话已终止，用户下次尝试发出 SQL 语句时会收到以下错误：

ORA-03135: connection lost contact

注：如果会话出现空闲超时，PMON 会话检测程序会自动终止会话，这可以使用概要文件或资源管理器来完成。

## 使用 SQL 解决锁冲突

可以使用 SQL 语句来确定阻塞会话并终止该会话。

①

```
SQL> select SID, SERIAL#, USERNAME  
      from V$SESSION where SID in  
      (select BLOCKING_SESSION from V$SESSION)
```

结果:

SID	SERIAL#	USERNAME
144	8982	HR

②

```
SQL> alter system kill session '144,8982' immediate;
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 使用 SQL 解决锁冲突

与在 Enterprise Manager 中执行的大多数其它任务一样，会话操作也可以通过发出 SQL 语句来完成。V\$SESSION 表包含所有已连接会话的详细信息。BLOCKING\_SESSION 中的值是阻塞会话的会话 ID。如果查询 SID 和 SERIAL#（其中 SID 与阻塞会话 ID 相匹配），就会得到执行 kill session 操作所需的信息。

注：可以使用数据库资源管理器自动注销阻塞其它会话的空闲会话。

# 死锁

事务处理 1		事务处理 2
<pre>UPDATE employees SET salary = salary * 1.1 WHERE employee_id = 1000;</pre>	9:00	<pre>UPDATE employees SET manager = 1342 WHERE employee_id = 2000;</pre>
<pre>UPDATE employees SET salary = salary * 1.1 WHERE employee_id = 2000;</pre>	9:15	<pre>UPDATE employees SET manager = 1342 WHERE employee_id = 1000;</pre>
ORA-00060: Deadlock detected while waiting for resource	9:16	

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 死锁

死锁是锁冲突的一种特殊情况。两个或更多会话等待已被其中另一会话锁定的数据时，就会发生死锁。因为每个会话都在等待另一个会话释放锁，所以任何一个会话都不能完成事务处理，也就不能解决冲突。

Oracle DB 会自动检测死锁并终止发生错误的语句。面对这种错误的适当做法是执行提交或回退，这样做会释放该会话中的其它所有锁，以便其它会话可继续完成其事务处理。

在幻灯片示例中，事务处理 1 必须提交或回退，才能更正检测到的死锁错误。如果执行提交，则必须重新提交第二次更新才能完成事务处理。如果执行回退，则必须同时重新提交这两个语句才能完成事务处理。

## 小测验

默认情况下，锁定机制采用细粒度行级锁定模式。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 1

## 小测验

发生死锁时，Oracle DB 会自动：

1. 等待 300 秒后终止两个会话
2. 终止一个会话中出错的一个语句
3. 终止两个会话中出错的两个语句
4. 默认情况下不采取任何操作，留给 DBA 处理

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案：2

## 小结

在本课中，您应该已经学会：

- 描述锁定机制以及 Oracle 如何管理数据并发处理
- 监视和解决锁定冲突

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 9 概览：管理数据和并发处理

本练习包含以下主题：

- 找出锁定冲突
- 解决锁定冲突

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

# 10

管理还原数据

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 说明 DML 和还原数据的生成
- 监视和管理还原数据
- 描述还原数据和重做数据之间的不同
- 配置还原保留期
- 保证还原保留期
- 使用还原指导

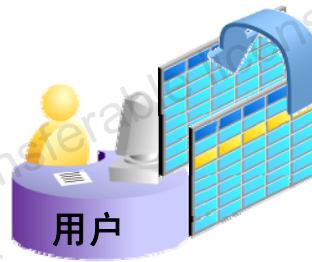
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 还原数据

还原数据是：

- 原始的、修改之前的数据副本
- 针对更改数据的每个事务处理而捕获
- 至少保留到事务处理结束
- 用于支持：
  - 回退操作
  - 读取一致性查询
  - Oracle 闪回查询、Oracle 闪回事务处理和 Oracle 闪回表
  - 从失败的事务处理中进行恢复



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 还原数据

当某个进程更改了数据库中的数据时，Oracle DB 会保存旧值（还原数据）。按数据修改前的原样存储数据。通过捕获还原数据可以回退未提交的数据。还原数据用于支持读取一致性和闪回查询。还原还可用于“倒回”（闪回）事务处理和表。

读取一致性查询提供与查询开始时的数据一致的结果。如果要成功完成读取一致性查询，原始信息必须仍作为还原信息存在。如果原始数据不再可用，则会收到“快照太旧”错误 (ORA-01555)。只要保留了还原信息，Oracle DB 就能通过重建数据来满足读取一致性查询的要求。

闪回查询有目的地查找过去某个时间存在的某个版本的数据。只要过去那个时间的还原信息仍存在，闪回查询就能成功完成。Oracle 闪回事务处理使用还原功能来创建补偿事务处理，以便回退事务处理及其相关事务处理。使用 Oracle 闪回表功能可将表恢复到特定的时间点。

还原数据也可用于从失败的事务处理中进行恢复。如果在用户决定提交或回退事务处理之前用户会话异常结束（可能因为网络错误或客户机故障），就会导致事务处理失败。当实例崩溃或您发出 SHUTDOWN ABORT 命令时，事务处理也可能会失败。

## 还原数据（续）

如果事务处理失败，则选择最安全的行为，即让 Oracle DB 撤销用户做出的所有更改，从而还原原始数据。

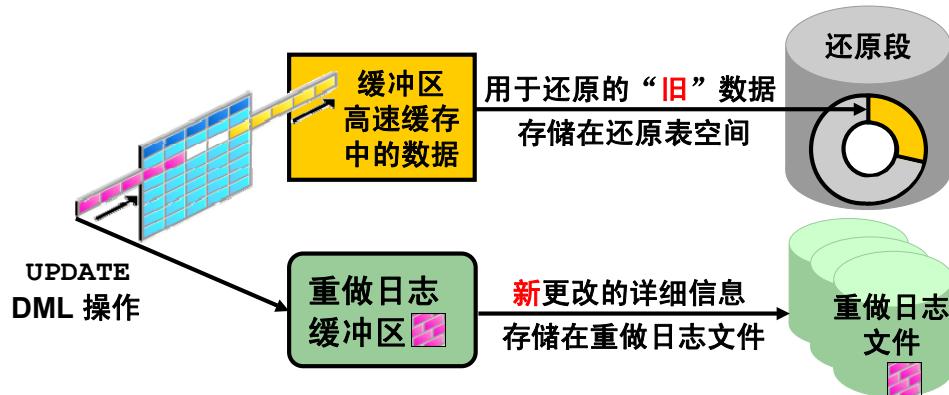
所有事务处理的还原信息将至少保留到事务处理结束，即使用以下方法之一结束事务处理时：

- 用户还原事务处理（回退事务处理）。
- 用户结束事务处理（提交事务处理）。
- 用户执行 DDL 语句，如 CREATE、DROP、RENAME 或 ALTER 语句。如果当前事务处理包含任何 DML 语句，则数据库首先提交该事务处理，然后在新的事务处理中执行并提交 DDL。
- 用户会话异常终止（回退事务处理）。
- 用户会话正常终止并退出（提交事务处理）。

保留的还原数据量及其保留时间取决于数据库活动量以及数据库配置。

**注：**Oracle 闪回事务处理利用联机重做日志来挖掘用于执行的相应还原 SQL。如果闪回事务处理调用中未提供事务处理开始时间，则只使用还原数据作为人为的时间边界，以此确定目标事务处理的重做挖掘开始时间。

## 事务处理和还原数据



- 每个事务处理仅分配一个还原段。
- 一个还原段可以同时服务多个事务处理。

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 事务处理和还原数据

事务处理开始时，为其分配还原段。在事务处理的整个生命期内，更改数据时，原始（更改之前）的值会复制到还原段。通过查看 V\$TRANSACTION 动态性能视图，可以了解哪些事务处理分配到哪些还原段。

还原段是由实例根据需要自动创建的专用段，用于支持事务处理。像所有段一样，还原段由区组成，区又由数据块组成。还原段可根据需要自动增长和收缩，它充当一个用于所分配的事务处理的循环存储缓冲区。

事务处理会填充其还原段中的区，直至完成了事务处理或占用了所有空间为止。如果填充完区之后还需要更多的空间，则事务处理将获取段中下一个区的空间。占用了所有区之后，事务处理会转回到第一个区或请求为还原段分配新区。

**注：**并行 DML 和 DDL 操作实际上可导致事务处理使用多个还原段。要了解关于并行 DML 执行的详细信息，请参阅《Oracle 数据库管理员指南》。

## 存储还原信息

还原信息存储在还原段中，还原段又存储在还原表空间中。

还原表空间：

- 仅用于还原段
- 有特殊的恢复注意事项
- 只能与单个实例相关联
- 在任意给定时间，一个给定的实例只能有一个表空间是当前可写还原表空间

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 存储还原信息

还原段只存在于特定形式的表空间中，这种形式的表空间称为“还原表空间”。（在还原表空间中无法创建其它段类型，如表。）

DBCA 可自动创建小文件还原表空间。用户还可以创建大文件还原表空间。但是，在同时执行大量短事务处理的大容量联机事务处理 (OLTP) 环境中，会在文件头上发生争用。存储于多个数据文件中的还原表空间可以解决这种潜在的问题。

尽管数据库可能有许多还原表空间，但是对于数据库中的任一实例，一次只能将一个还原表空间指定为当前还原表空间。

还原段由 SYS 自动创建并始终归其所有。由于还原段充当循环缓冲区，因此每个段最少包含两个区。默认的最大区数取决于数据库块大小，但是此值非常大（块大小为 8 KB 时区数为 32,765）。

还原表空间是永久的、本地管理的表空间（具有自动区分配）。它们由数据库自动进行管理。

由于需要有还原数据才能从失败的事务处理（如实例崩溃时可能会导致事务处理失败）中进行恢复，所以还原表空间只在实例处于 MOUNT 状态时才可恢复。还原表空间的恢复注意事项将在“执行数据库恢复”一课中介绍。

## 还原数据与重做数据

	还原	重做
记录	如何还原更改	如何重现更改
用于	回退、读取一致性、闪回	前滚数据库更改
存储于	还原段	重做日志文件
避免	在多用户系统中出现读取不一致	数据丢失



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 还原数据与重做数据

还原数据和重做数据初看很相似，但是二者的作用却截然不同。如果需要还原更改，则需要还原数据，而且为了保持读取一致性和执行回退，也需要还原数据。在因某种原因而丢失了所做更改时，如果需要再次执行更改，则需要用到重做数据。还原块更改也写入重做日志。

提交过程需要验证在事务处理中所做的更改是否已写入重做日志文件。此重做日志文件会永久保存在磁盘上，而不是保存在内存中。此外，重做日志文件通常是多路复用的。因此，磁盘上有重做数据的多个副本。虽然更改可能尚未写入实际存储表块的数据文件，但只要写入到持久重做日志，就足以确保数据库的一致性。

假设在已提交的更改反映到数据文件之前刚好发生断电。因为事务处理已提交，所以这种情况不会导致问题。当系统再次启动时，就能够前滚断电时尚未反映到数据文件中的任何重做记录。

## 管理还原

自动还原管理:

- 在专用还原表空间中对还原数据和空间进行完全自动的管理
- 用于所有会话
- 在 AUTOEXTEND 表空间中进行自动优化，以满足长时间运行的查询的需求
- 在固定大小的表空间中进行自动优化，以实现最佳的保留期

用来支持闪回操作的 DBA 任务:

- 配置还原保留期
- 将还原表空间更改为固定大小
- 避免空间错误和“快照太旧”错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 管理还原

Oracle DB 提供了“自动还原管理”功能，这是一种完全自动的机制，用于在一个专用的还原表空间中管理所有会话的还原信息和空间。系统自动优化自身，从而尽可能提供最佳的还原信息保留期。更准确地说，对于自动扩展的表空间，会将还原保留期优化为略长于运行时间最长的活动查询。对于固定大小的还原表空间，数据库会进行动态优化，尽可能获得最佳保留时间。

自动还原管理是 Oracle Database 11g（和更高版本）中的默认功能。也支持手动还原管理，以便与 Oracle8i 及较早版本向后兼容，但是 DBA 需要进行更多的交互操作。在手动还原管理模式下，通过回退段而不是还原表空间来管理还原空间。

**注：**Oracle 强烈建议您使用自动还原管理。

尽管在默认情况下，Oracle DB 自动管理还原数据和空间，但如果数据库在使用闪回操作，则可能需要执行某些任务。管理还原数据和空间时，应防止空间错误、使用过多的空间和“快照太旧”错误。

## 配置还原保留时间

UNDO\_RETENTION 指定已提交的还原信息要保留多长时间（秒）。仅在以下情况下才必须设置此参数：

- 还原表空间启用了 AUTOEXTEND 选项
- 需要设置 LOB 的还原保留时间
- 需要保证保留时间



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 配置还原保留时间

UNDO\_RETENTION 初始化参数指定还原保留时间的下限阈值（秒）。可将自动扩展的还原表空间的最短还原保留期设置为预期的最长闪回操作所需的时间。对于自动扩展的还原表空间，系统会至少将还原数据保留此参数指定的时间，而且还会自动优化还原保留期，以满足查询的还原要求。但是，此自动优化的保留期可能不足以执行闪回操作。

对于固定大小的还原表空间，系统会根据还原表空间大小和使用历史记录进行自动优化，尽可能获得最佳还原保留期；除非启用了保留时间保证，否则系统会忽略 UNDO\_RETENTION。因此，进行自动还原管理时，幻灯片中列出的三种情况都使用 UNDO\_RETENTION 设置。

对于这三种以外的情况，会忽略此参数。

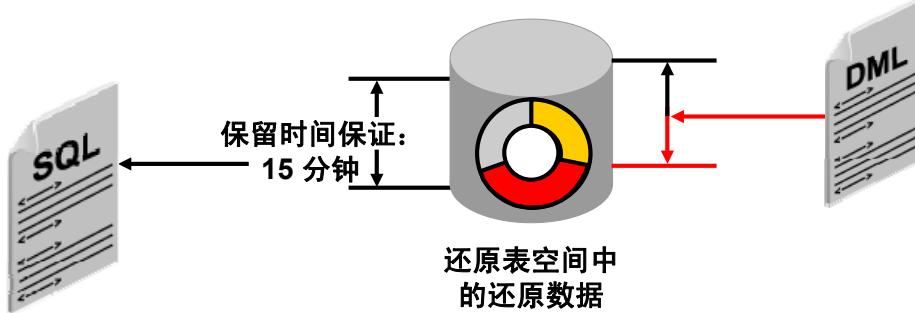
## 配置还原保留时间（续）

还原信息分为三类：

- **未提交的还原信息（活动）**：用来支持当前运行的事务处理；如果用户要回退事务处理或事务处理失败时，需要用到这类信息。绝对不会覆盖未提交的还原信息。
- **提交的还原信息（未过期）**：不再需要用来支持运行的事务处理，但是为了符合还原保留间隔，仍然要用到这类信息。这也称为“未过期”还原信息。只要不会因缺少空间而导致活动事务处理失败，就会保留提交的还原信息。
- **过期的还原信息（过期）**：不再需要用来支持运行的事务处理。活动事务处理需要空间时会覆盖过期的还原信息。

## 保证还原保留时间

```
SQL> ALTER TABLESPACE undotbs1 RETENTION GUARANTEE;
```



运行时间不超过 15 分钟  
的 SELECT 语句始终  
可以成功执行。

如果事务处理生成的还原  
数据超过可用空间，  
事务处理就会失败。

注：本示例基于 900 秒（15 分钟）的 UNDO\_RETENTION 设置。

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 保证还原保留时间

默认还原行为是覆盖提交的、尚未过期的事务处理的还原信息，而不是使活动事务处理因缺少还原空间而失败。

此行为在设置了保留时间保证时会有所改变。设置了保留时间保证时，即使会导致事务处理失败，也仍会强制执行还原保留时间设置。

RETENTION GUARANTEE 是表空间属性而不是初始化参数。此属性只可使用 SQL 命令行语句来更改。将还原表空间改为采用保留时间保证设置的语法是：

```
SQL> ALTER TABLESPACE undotbs1 RETENTION GUARANTEE;
```

要将设置了保留时间保证的还原表空间返回到其常规设置，请使用以下命令：

```
SQL> ALTER TABLESPACE undotbs1 RETENTION NOGUARANTEE;
```

保留时间保证设置仅适用于还原表空间。尝试对非还原表空间设置保留时间保证会产生以下错误：

```
SQL> ALTER TABLESPACE example RETENTION GUARANTEE;
ERROR at line 1:
ORA-30044: 'Retention' can only specified for undo tablespace
```

## 将还原表空间改为固定大小

原因：

- 支持闪回操作
- 限制表空间增长

工作流：

1. 运行正常工作量。
2. 自动优化机制确定所需的最小大小。
3. 可以选择使用还原指导，该指导可计算为了满足未来增长而需要的大小。
4. 可以选择将还原表空间改为固定大小。



**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 将还原表空间改为固定大小

可能出于以下两个原因，需要将还原表空间改为固定大小：支持闪回操作（预期将来会使用还原功能）或防止表空间增长过大。

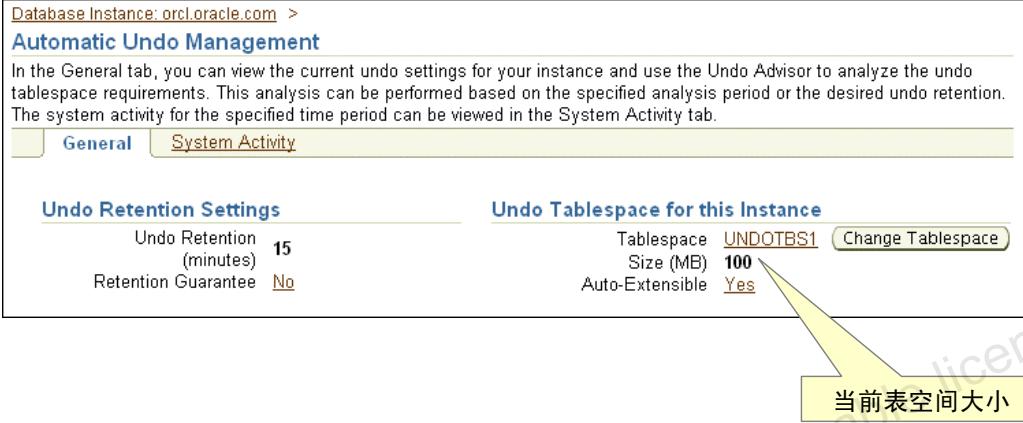
如果决定将还原表空间改为固定大小，则必须选择一个足够大的大小以避免以下两类错误：

- DML 失败（因为空间不够大，无法存储新事务处理的还原数据）
- “快照太旧” 错误（因为没有足够的还原数据来实现读取一致性）

Oracle 建议您运行正常的满负荷工作量，使还原表空间增长到所需的最小大小。自动收集的统计信息包括运行时间最长的查询的持续时间和还原数据生成速率。对于不执行闪回操作的系统以及将来不打算对其执行运行时间更长的查询的系统，建议根据这些统计信息计算最小还原表空间大小。

对于运行时间更长的查询和闪回操作，可以使用还原指导来输入所需的还原保留时间。

## 一般还原信息



Database Instance: orcl.oracle.com >

**Automatic Undo Management**

In the General tab, you can view the current undo settings for your instance and use the Undo Advisor to analyze the undo tablespace requirements. This analysis can be performed based on the specified analysis period or the desired undo retention. The system activity for the specified time period can be viewed in the System Activity tab.

<b>General</b>	<b>System Activity</b>								
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <th style="width: 30%;">Undo Retention Settings</th> <th style="width: 70%;">Undo Tablespace for this Instance</th> </tr> <tr> <td style="padding: 2px;">Undo Retention (minutes) <b>15</b></td> <td style="padding: 2px;">Tablespace <b>UNDOTBS1</b> <span style="float: right;">Change Tablespace</span></td> </tr> <tr> <td style="padding: 2px;">Retention Guarantee <b>No</b></td> <td style="padding: 2px;">Size (MB) <b>100</b></td> </tr> <tr> <td></td> <td style="padding: 2px;">Auto-Extensible <b>Yes</b></td> </tr> </table>		Undo Retention Settings	Undo Tablespace for this Instance	Undo Retention (minutes) <b>15</b>	Tablespace <b>UNDOTBS1</b> <span style="float: right;">Change Tablespace</span>	Retention Guarantee <b>No</b>	Size (MB) <b>100</b>		Auto-Extensible <b>Yes</b>
Undo Retention Settings	Undo Tablespace for this Instance								
Undo Retention (minutes) <b>15</b>	Tablespace <b>UNDOTBS1</b> <span style="float: right;">Change Tablespace</span>								
Retention Guarantee <b>No</b>	Size (MB) <b>100</b>								
	Auto-Extensible <b>Yes</b>								

当前表空间大小

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 一般还原信息

在 Enterprise Manager 中，选择“Server > Automatic Undo Management（服务器 > 自动还原管理）”。

有以下两页：“General（一般信息）”和“System Activity（系统活动）”。在“General（一般信息）”页的顶部，可以看到“Undo Retention Settings（还原保留时间设置）”和有关“Undo Tablespace for this Instance（该实例的还原表空间）”的信息。

## 使用还原指导

**Undo Advisor: Undo Retention and Undo Tablespace Sizing Advice**

Undo retention is the length of time that undo data is retained in the undo tablespaces. Undo data must be retained for the length of the longest running query, the longest running transaction, and the longest flashback duration (except for Flashback Database). The undo tablespace should be sized large enough to hold the undo generated by the database during the undo retention period. Note that the undo retention parameter is also used as the retention value for LOB columns.

**Analysis Period**

Analysis Time Period	Last One Hour
Desired Undo Retention	<input checked="" type="radio"/> Automatically chosen based on longest query in analysis period <input type="radio"/> Specified manually to allow for longer duration queries or flashback
Duration	minutes
<a href="#">Run Analysis</a>	

**Analysis Results**

Selected Analysis Time Period	Jun 17, 2009 11:04:47 AM GMT+07:00 To J PM GMT+07:00	<a href="#">Edit Undo Tablespace</a>
Minimum Required Undo Tablespace Size	10 (MB)	<b>15</b>
Recommended Undo Tablespace Size (MB)	15	<b>15</b>
<input checked="" type="checkbox"/> <b>TIP</b> Recommended size is three times the minimum size to allow for workload fl		
Potential Problems	<b>No Problem Found</b>	
Recommendations	<b>No Recommendation</b>	

[Show Graph](#)

Required Tablespace Size (MB)

Undo Retention (minutes)

Auto-tuned Undo Retention (16, 10)

New (81955, 15279)

Best Possible Undo Retention (172,997, 32251)

Clicking a point on the graph updates the Duration field and the Minimum Required Undo Tablespace Size field.

**ORACLE**

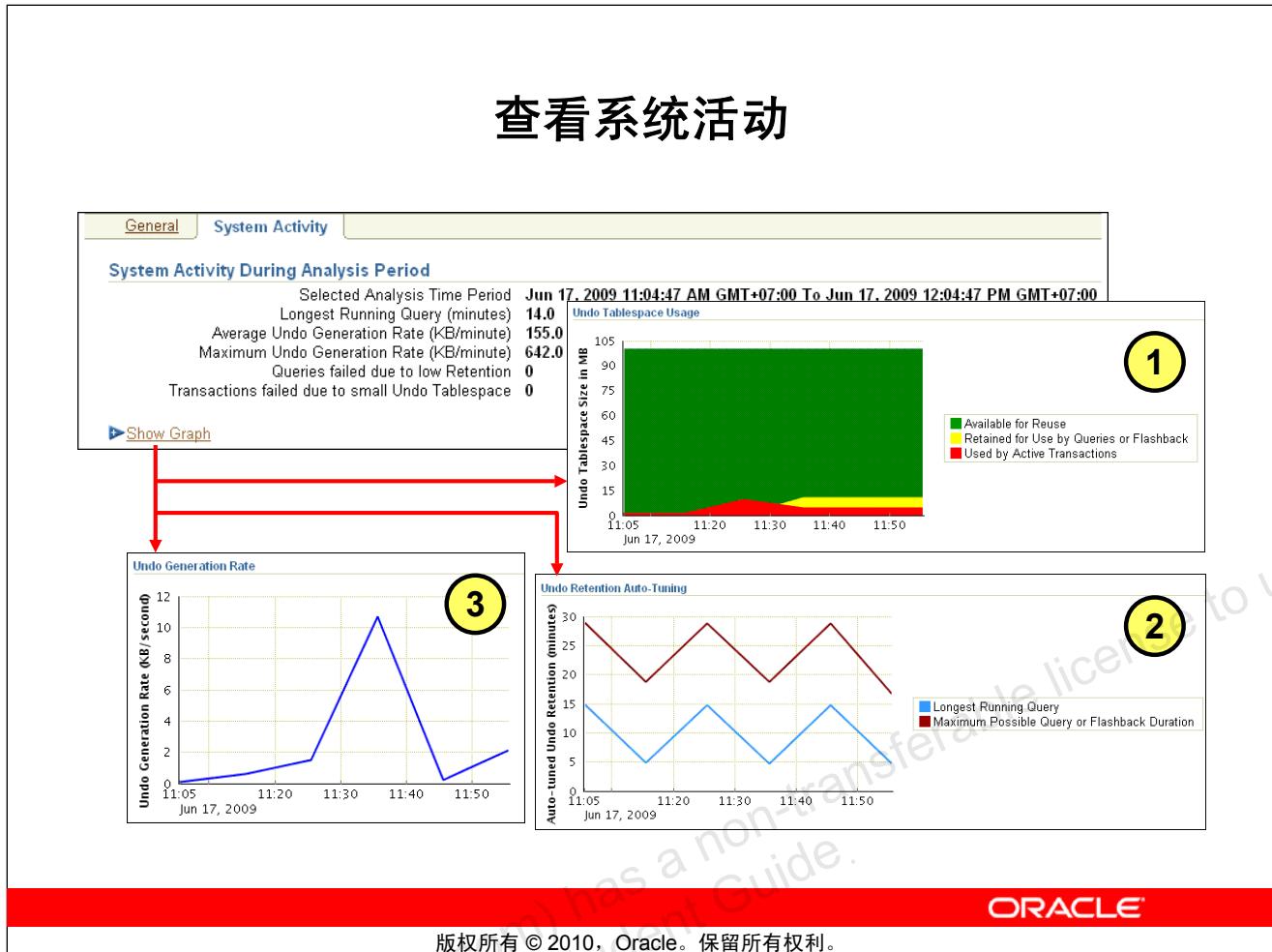
版权所有 © 2010, Oracle。保留所有权利。

## 使用还原指导

在“General Undo（一般还原）”页的中间部分，可以访问还原指导。通过它可估计满足指定的还原保留时间所需要的还原表空间大小。

该指导的分析区域会显示支持保留期所需的表空间大小。可以通过单击图上的某个点来查看支持选定期间所需要的表空间大小。

单击“Edit Undo Tablespace（编辑还原表空间）”按钮，然后在“Datafile（数据文件）”部分单击“Edit（编辑）”，将还原表空间改为固定大小。



### 查看系统活动

页面顶部将显示所选期间的系统活动。

在此部分的下方，有三个图形：

- 1. Undo Tablespace Usage** (还原表空间使用情况)：显示一月中某日的表空间大小 (以 MB 为单位)
- 2. Undo Retention Auto-Tuning** (还原保留时间自动优化)：显示一月中某日的优化还原保留时间 (以分钟为单位)
- 3. Undo Generation Rate** (还原生成速率)：显示某日的还原生成速率 (以 KB/秒为单位)

## 小测验

要保证 15 分钟以下的所有查询可找到实现读取一致性所需的还原数据，所要做的只是将 UNDO\_RETENTION 参数设为 15 分钟。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小测验

哪项陈述与还原数据无关？

1. 提供如何还原更改的记录
2. 用于回退、读取一致性和闪回
3. 只存储在内存中，不写入磁盘
4. 防止在多用户系统中出现读取不一致

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 3

## 小结

在本课中，您应该已经学会：

- 说明 DML 和还原数据的生成
- 监视和管理还原数据
- 描述还原数据和重做数据之间的不同
- 配置还原保留期
- 保证还原保留期
- 使用还原指导

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 10 概览：管理还原段

本练习包含以下主题：

- 查看系统活动
- 计算支持 48 小时保留间隔所需的还原表空间大小
- 修改还原表空间以支持 48 小时保留间隔

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



# 11

## 实施 Oracle DB 审计

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

学完本课后，应能完成以下工作：

- 说明 DBA 负责的安全和审计工作
- 启用标准数据库审计
- 指定审计选项
- 复查审计信息
- 维护审计线索



版权所有 © 2010, Oracle。保留所有权利。

### 课程目标

本课是学习 Oracle 安全性知识的入门课程。补充信息包括在以下文档中：

- 《Oracle Database Concepts》
- 《Oracle 数据库管理员指南》
- 《Oracle 数据库安全性指南》

下列课程中提供了进一步的培训：

- Oracle Database 11g: 数据库管理 – 课堂练习 II
- Oracle Database 11g: 安全性

## 责任分离

- 具有 DBA 权限的用户必须是可信任的。
  - 滥用信任
  - 用审计线索保护受信任位置
- 必须共同分担 DBA 责任。
- 绝对不要共享帐户。
- DBA 和系统管理员必须由不同的人员担任。
- 分离操作员与 DBA 的责任。



版权所有 © 2010, Oracle。保留所有权利。

### 责任分离

以下是满足责任分离的主要要求。

**DBA 必须是可信任的:** 很难限制某个 DBA 去做什么。为了完成工作, DBA 需要很高的权限。DBA 是受信任的职位, 因此必须接受全面检查。即使是受信任的 DBA 也必须承担责任。请考虑以下因素:

- **滥用信任:** DBA 可能会滥用 DBA\_USERS 视图中的加密密码。
- **用审计线索保护受信任位置:** 谨慎实施审计且遵守准则之后, 审计线索会指出特定人员没有违反规程, 也没有破坏性行为。如果有恶意用户试图让人怀疑受信任的用户, 设计良好的审计线索可捕获该行为。

**Oracle Database Vault:** 在下面的情况下, 可以使用 Oracle Database Vault 选件: 数据库必须强制执行责任分离, 或者不允许 DBA 查看某些或所有数据库方案中的数据。

## 数据库安全性

安全的系统可确保所包含数据的机密性。有以下几方面的安全性功能：

- 限制对数据和服务的访问
- 验证用户
- 监视可疑活动



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据库安全性

Oracle Database 11g 提供了业界最佳的安全系统框架。但是，要让这个框架起作用，数据库管理员必须遵循最佳实践并持续监视数据库活动。

#### 限制对数据和服务的访问

不是所有用户都应对所有数据具有访问权。根据数据库中存储的内容，可按业务需要、客户期望以及日益增加的法律限制条款来强制实施有限制的访问。必须保护信用卡信息、医疗保健数据、身份识别信息等，使之免受未授权访问的侵害。Oracle DB 通过提供细粒度的授权控制来限制数据库访问。限制访问必须包括应用最少权限原则。

## 数据库安全性（续）

### 验证用户

为了强制对敏感数据实施访问控制，系统必须首先知道是谁尝试访问数据。如果验证机制有漏洞，就会导致所有其它安全预防措施变得无用。最基本的用户验证方式是要求用户提供知道的验证信息，如口令。如果可以保证口令遵循简单规则，就可极大地增强系统的安全性。比较严格的验证方法包括要求用户提供掌握的某些验证信息，如令牌或公共密钥基础结构 (PKI) 证书。更严格的验证方法是，通过诸如指纹、虹膜、骨组织模式等唯一生物学特征来识别用户。Oracle DB 支持通过高级安全选件来使用高级验证技术（如基于令牌、生物学和证书的身份识别技术）。为了防止有人钻验证漏洞，必须锁定当前未使用的用户帐户。

### 监视可疑活动

即使经过授权和验证的用户有时也会钻系统漏洞。为了查到信息失窃的原因，第一步就是要找出不寻常的数据库活动，如某个雇员突然开始查询大量的信用卡信息、研究结果或其他敏感信息。为了跟踪用户活动和确定可疑活动的变化趋向，Oracle DB 提供了很多审计工具。

## 监视合规性

监视或审计是安全过程不可缺少的一部分。

请复查下列各项：

- 强制性审计
- 标准数据库审计
- 基于值审计
- 细粒度审计 (FGA)
- SYSDBA (和 SYSOPER) 审计



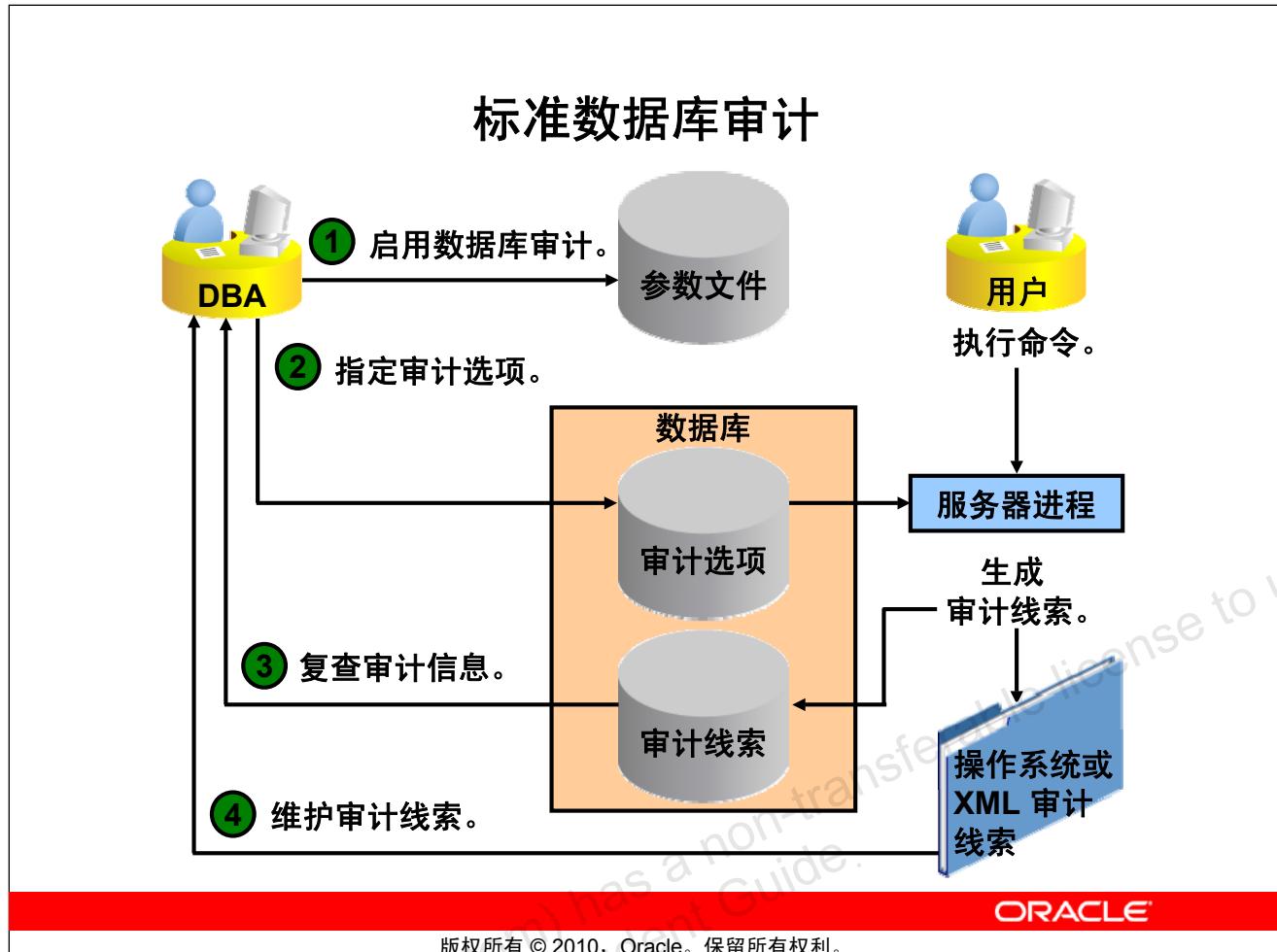
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 监视合规性

审计意味着捕获并存储系统上所发生情况的信息，这会增加系统必须执行的工作量。审计必须有重点，以便只捕获有意义的事件。如果审计重点设置适当，则会最大程度地减少对系统性能的影响。如果审计重点设置不当，则会对系统性能产生明显的影响。

- **强制性审计：**不管其它审计选项或参数如何设置，所有 Oracle DB 都会审计特定的操作。由于数据库需要记录诸如授权用户连接等数据库活动，所以存在强制性审计日志。
- **标准数据库审计：**通过使用 AUDIT\_TRAIL 初始化参数在系统级别启用。启用审计之后，选择要审计的对象和权限，并使用 AUDIT 命令设置审计属性。
- **基于值审计：**扩展了标准数据库审计的功能，不仅会捕获发生的审计事件，还会捕获插入、更新或删除的实际值。基于值审计是通过数据库触发器实施的。
- **细粒度审计 (FGA)：**扩展了标准数据库审计的功能，从而可捕获发出的实际 SQL 语句，而不仅仅是发生事件的情况。
- **SYSDBA (和 SYSOPER) 审计：**将 DBA 与审计者或安全管理员的审计责任分离开，审计者或安全管理员在操作系统审计线索中负责监视 DBA 的活动。



## 标准数据库审计

要使用数据库审计，必须先将静态 AUDIT\_TRAIL 参数设置为指向审计记录的存储位置。这样做可启用数据库审计。启用数据库审计并指定审计选项（登录事件、行使的系统和对象权限或使用的 SQL 语句）后，数据库开始收集审计信息。

如果将 AUDIT\_TRAIL 设置为 OS，审计记录存储在操作系统的审计系统中。在 Windows 环境下，审计记录存储在事件日志中。在 UNIX 或 Linux 环境下，审计记录存储在使用 AUDIT\_FILE\_DEST 参数指定的文件中。

如果将 AUDIT\_TRAIL 参数设置为 DB 或 DB\_EXTENDED，您可以在 DBA\_AUDIT\_TRAIL 视图（SYS 方案的一部分）中查看审计记录。

如果 AUDIT\_TRAIL 设置为 XML 或 XML\_EXTENDED，审计记录会写入 AUDIT\_FILE\_DEST 参数指向的目录中的 XML 文件。使用 V\$XML\_AUDIT\_TRAIL 视图可查看此目录中的所有 XML 文件。

维护审计线索是一项重要的管理任务。审计线索可能会迅速地增长，具体取决于审计选项的审计重点。如果维护不当，审计线索会创建过多的记录，以至影响系统的性能。审计开销与生成的记录号直接相关。

## 配置审计线索

可使用 AUDIT\_TRAIL 启用数据库审计。

The parameter values listed here are from the SPFILE +DATA/orcl/spfileorcl.ora

Name	Basic	Dynamic	Category
audit	All	All	All

Filter on a name or partial name

Apply changes in SPFile mode to the current running instance(s). For static parameters, you must restart the database.

Select	Name	Help	Revisions	Value	Comments	Type	Basic	Dynamic	Category
<input checked="" type="radio"/>	audit_file_dest	<a href="#">?</a>		/1/app/oracle/admin/orcl/audit		String	<input checked="" type="checkbox"/>		Security and Auditing
<input checked="" type="radio"/>	audit_sys_operations	<a href="#">?</a>		Unspecified		Boolean			Security and Auditing
<input checked="" type="radio"/>	audit_syslog_level	<a href="#">?</a>				String			Miscellaneous
<input checked="" type="radio"/>	audit_trail	<a href="#">?</a>		XML		String			Security and Auditing

**ALTER SYSTEM SET AUDIT\_TRAIL='XML' SCOPE=SPFILE;**

请在修改静态初始化参数之后重新启动数据库。

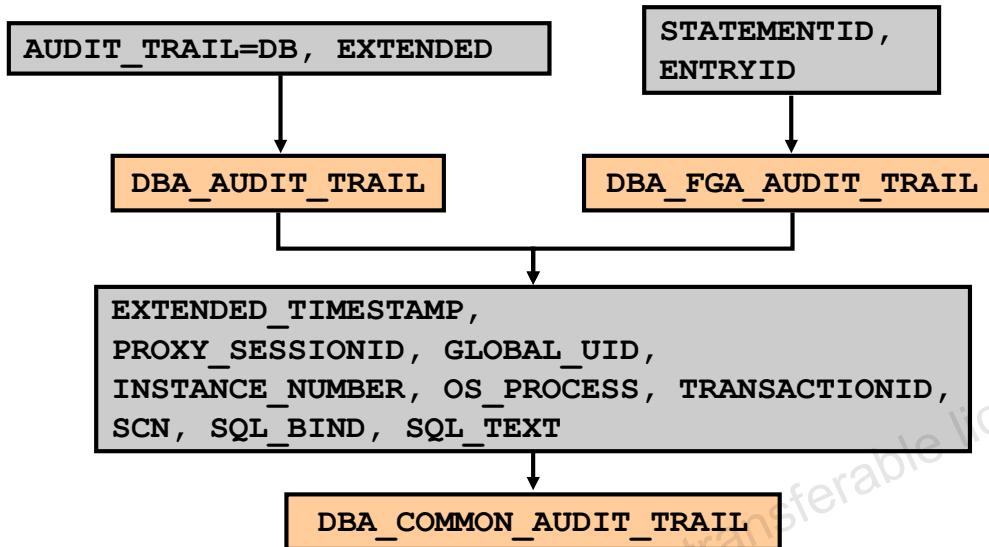
ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 配置审计线索

可以在 Enterprise Manager 中使用 “Initialization Parameters (初始化参数) ” 页或者在 SQL\*Plus 中使用 ALTER SYSTEM SET 命令来设置 AUDIT\_TRAIL 参数。因为这是一个静态参数，所以您必须重启数据库，更改才能生效。如果数据库是使用 Database Configuration Assistant (DBCA) 创建的，默认情况下，audit\_trail 参数设置为 DB。当 AUDIT\_TRAIL 设置为 DB 时，默认的行为是将审计线索记录到数据库的 AUD\$ 表中。对于大多数站点而言，这类审计不会对数据库性能造成太大影响。Oracle 建议使用操作系统审计线索文件。如果数据库是手动创建的（使用 CREATE DATABASE 命令），默认将 AUDIT\_TRAIL 设置为 NONE。

## 统一审计线索



**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### 统一审计线索

Oracle DB 进行标准审计和细粒度审计时跟踪相同的字段，这使您可以轻松分析数据库活动。为实现这一点，标准审计线索和细粒度审计线索包含了彼此互补的属性。

通过标准审计收集的额外信息包括：

- 系统更改号 (SCN)，记录对系统的每一项更改。
- 用户执行的确切 SQL 文本及与 SQL 文本一起使用的绑定变量。只有已指定 **AUDIT\_TRAIL=DB, EXTENDED** 的情况下，这些列才会出现。

通过细粒度审计收集的额外信息包括：

- 每个审计记录的序列号。
- 将源自一条语句的多个审计条目联系起来的语句编号。

公共属性包括：

- 用全球标准时间 (UTC) 表示的全球时间戳。这个字段在监视不同地理位置和不同时区中的数据库时特别有用。
- 每个 Real Application Cluster (RAC) 实例的唯一实例编号。
- 用于将一个事务处理的审计记录组成一组的事务处理标识符。

**DBA\_COMMON\_AUDIT\_TRAIL** 视图中组合了标准审计日志记录和细粒度审计日志记录。

## 指定审计选项

- SQL 语句审计:

```
AUDIT table;
```

- 系统权限审计（非重点和重点）：

```
AUDIT select any table, create any trigger;
AUDIT select any table BY hr BY SESSION;
```

- 对象权限审计（非重点和重点）：

```
AUDIT ALL on hr.employees;
AUDIT UPDATE,DELETE on hr.employees BY ACCESS;
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 指定审计选项

**SQL 语句审计:** 幻灯片中显示的语句可审计影响表的任何数据定义语言 (DDL) 语句，包括 CREATE TABLE、DROP TABLE 和 TRUNCATE TABLE 等。可按用户名或者按成功或失败来设置 SQL 语句审计的重点：

```
SQL> AUDIT TABLE BY hr WHENEVER NOT SUCCESSFUL;
```

**系统权限审计:** 可用来审计行使的任何系统权限（如 DROP ANY TABLE）。可按用户名或者成功或失败设置审计的重点。默认情况下，审计设置为 BY ACCESS。每次行使经审计的系统权限时，都会生成一条审计记录。可选择使用 BY SESSION 子句将这些记录组成一组，以便每个会话只生成一条记录。（这样，如果一个用户针对另一个用户的表发出了多条更新语句，则只收集一条审计记录。）请考虑使用 BY SESSION 子句来限制由于审计系统权限而对性能和存储产生的影响。

**对象权限审计:** 可用来审计关于表、视图、过程、序列、目录和用户定义数据类型的操作。这种审计类型可按成功或失败设置审计的重点，而且可以按会话或访问权限分组。与系统权限审计不同，默认情况下，对象权限审计按会话分组。如果要为每个操作分别生成一条审计线索记录，必须显式指定 BY ACCESS。

# 默认审计

默认审计的权限		
ALTER ANY PROCEDURE	CREATE ANY LIBRARY	GRANT ANY PRIVILEGE
ALTER ANY TABLE	CREATE ANY PROCEDURE	GRANT ANY ROLE
ALTER DATABASE	CREATE ANY TABLE	DROP ANY PROCEDURE
ALTER PROFILE	CREATE EXTERNAL JOB	DROP ANY TABLE
ALTER SYSTEM	CREATE PUBLIC DATABASE LINK	DROP PROFILE
ALTER USER	CREATE SESSION	DROP USER
AUDIT SYSTEM	CREATE USER	EXEMPT ACCESS POLICY
CREATE ANY JOB	GRANT ANY OBJECT PRIVILEGE	

默认审计的语句
SYSTEM AUDIT BY ACCESS
ROLE BY ACCESS

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 默认审计

在 Oracle Database 11g 中启用审计时，将默认审计那些对安全性非常重要的特定权限和语句。系统将针对所有用户，按成功、失败以及访问来审计幻灯片中列出的这些权限和语句。

**Enterprise Manager 审计页**

**Security**

- Users
- Roles
- Profiles
- Audit Settings**
- Transparent Data Encryption

**Audit Settings**

**Configuration**

Audit Trail	DB
Audit SYS User Operations	FALSE
Audit File Directory	/u01/app/oracle/admin/orcl/adump
Audit File Directory value is effective only when Audit Trail is set to "OS" or "XML".	

**Audit Trails**

Database Audit Trail	<a href="#">Audited Failed Logins</a>
	<a href="#">Audited Privileges</a>
	<a href="#">Audited Objects</a>
Operating System Audit Trail	<a href="#">View OS Audit Trails</a>

**Default Options For Future Audited Objects** [0](#)

**Audited Privileges (23)** [Audited Objects \(0\)](#) [Audited Statements \(2\)](#)

Privilege	Privilege	User	Proxy	Success	Failure
<input type="checkbox"/>	DROP PROFILE			BY ACCESS	BY ACCESS
<input type="checkbox"/>	ALTER ANY TABLE			BY ACCESS	BY ACCESS
<input type="checkbox"/>	ALTER SYSTEM			BY ACCESS	BY ACCESS
<input type="checkbox"/>	ALTER DATABASE			BY ACCESS	BY ACCESS
<input type="checkbox"/>	DROP USER			BY ACCESS	BY ACCESS

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## Enterprise Manager 审计页

在 Database Control 主页中，通过单击“Server（服务器）”选项卡，然后单击“Security（安全性）”区域中的“Audit Settings（审计设置）”链接，可访问“Audit（审计）”页。“Audit（审计）”页上包含下列区域：

- Configuration（配置）**：显示当前的配置参数值，包含用于编辑参数值的链接
- Audit Trails（审计线索）**：可轻松访问已收集的审计信息

可使用以下选项卡页来设置和取消设置审计选项：

- Audited Privileges（审计的权限）**：显示审计的权限
- Audited Objects（审计的对象）**：显示审计的对象
- Audited Statements（审计的语句）**：显示审计的语句

# 使用和维护审计信息

The screenshot shows the 'Audit Trails' interface. In the 'Database Audit Trail' section, the 'Audited Objects' link is highlighted with a red box and a red arrow points from it to the 'Audited Objects' section of the 'Audited Objects' page.

**Audited Objects**

**Hide SQL**

```
SELECT 'OWNER', 'OBJ_NAME', 'USERNAME', 'ACTION_NAME', 'TIMESTAMP' FROM 'SYS'.'DBA_AUDIT_OBJECT'
ORDER BY extended_timestamp desc
```

Schema	Object Name	User Name	Action	Time
INVENTORY	PRODUCT_MASTER	DBA1	ALTER TABLE	2008-08-13 22:47:56.0
INVENTORY	PRODUCT_ON_HAND	DBA1	CREATE TABLE	2008-08-13 16:45:49.0

请在不使用审计选项时禁用审计选项。

**Confirmation**

Are you sure you want to remove the 4 selected audited objects?

The audited statements you remove will no longer be audited on the objects.

**Hide SQL**

```
NOAUDIT COMMENT ON HR.EMPLOYEES
NOAUDIT INDEX ON HR.EMPLOYEES
NOAUDIT LOCK ON HR.EMPLOYEES
NOAUDIT RENAME ON HR.EMPLOYEES
```

ORACLE

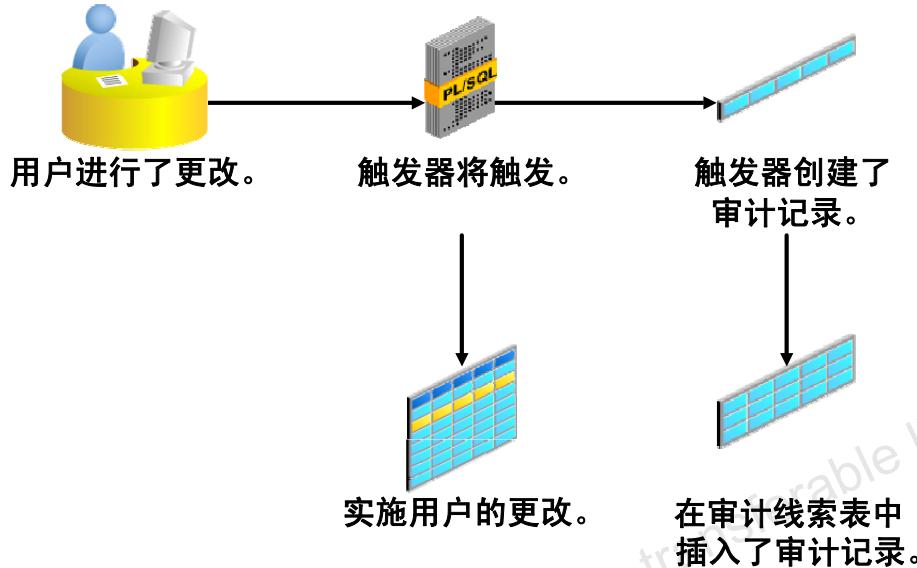
版权所有 © 2010, Oracle。保留所有权利。

## 使用和维护审计信息

### 最佳实践提示

审计会造成性能下降，其幅度与审计线索的写入数量成比例。要定制满足站点需求的审计选项，可只启用那些满足安全策略所需的选项。设置审计重点，以减少审计线索条目的数量。

## 基于值审计



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 基于值的审计

进行数据库审计时会记录审计对象中发生的插入、更新和删除操作，但是不会捕获更改的实际值。要扩展数据库审计，可使用基于值的审计，利用数据库触发器（事件驱动的 PL/SQL 构造）来捕获更改的值。

用户在连接了相应触发器的表中插入、更新或删除数据时，触发器在后台将审计信息复制到包含审计信息的表中。因为审计触发器代码在每次插入、更新或删除操作发生时都必须执行，所以与标准数据库审计相比，使用基于值的审计时，性能下降幅度比较大。性能下降幅度取决于触发器代码的效率。只在标准数据库审计捕获的信息不足的情况下，才使用基于值的审计。

基于值的审计由用户或第三方代码实施。Oracle DB 提供了可用来构建基于值的审计系统的 PL/SQL 构造。

## 基于值的审计（续）

基于值的审计的关键部分是审计触发器，这是一个单纯为了捕获审计信息而构造的 PL/SQL 触发器。

以下是一个审计触发器的典型示例：

```
CREATE OR REPLACE TRIGGER system.hrsalary_audit
    AFTER UPDATE OF salary
    ON hr.employees
    REFERENCING NEW AS NEW OLD AS OLD
    FOR EACH ROW
BEGIN
    IF :old.salary != :new.salary THEN
        INSERT INTO system.audit_employees
        VALUES (sys_context('userenv','os_user'), sysdate,
        sys_context('userenv','ip_address'),
        :new.employee_id ||
        ' salary changed from'||:old.salary||
        ' to'||:new.salary);
    END IF;
    END;
    /
```

这个触发器将审计的重点设置为捕获 hr.employees 表薪水列的更改。更新某行后，触发器就会检查薪水列。如果新旧薪水不相等，则触发器会在 audit\_employees 表（通过在 SYSTEM 方案中单独执行一项操作而创建）中插入一条审计记录。审计记录中包括了用户名、执行更改的 IP 地址、标识所更改记录的主键及更改的实际薪水值。

如果标准数据库审计收集的数据不足，还可使用数据库触发器来捕获关于用户连接的信息。通过使用登录触发器，管理员可以捕获用来标识连接到数据库的用户的 data。示例中包括下列各项：

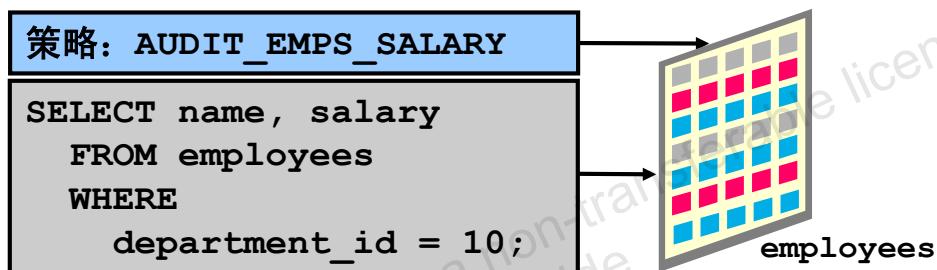
- 登录人员的 IP 地址
- 用于连接到实例的程序名的前 48 个字符
- 用于连接到实例的终端名

有关用户参数的完整列表，请参阅《Oracle 数据库 SQL 参考》中的“SYS\_CONTEXT”一节。

在许多情况下，会使用细粒度审计 (FGA) 功能，而不会使用基于值的触发器。

## 细粒度审计

- 根据内容监视数据访问
- 审计 SELECT、INSERT、UPDATE、DELETE 和 MERGE
- 可链接到表或视图中的一列或多列
- 可以执行某个过程
- 使用 DBMS\_FGA 程序包进行管理



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

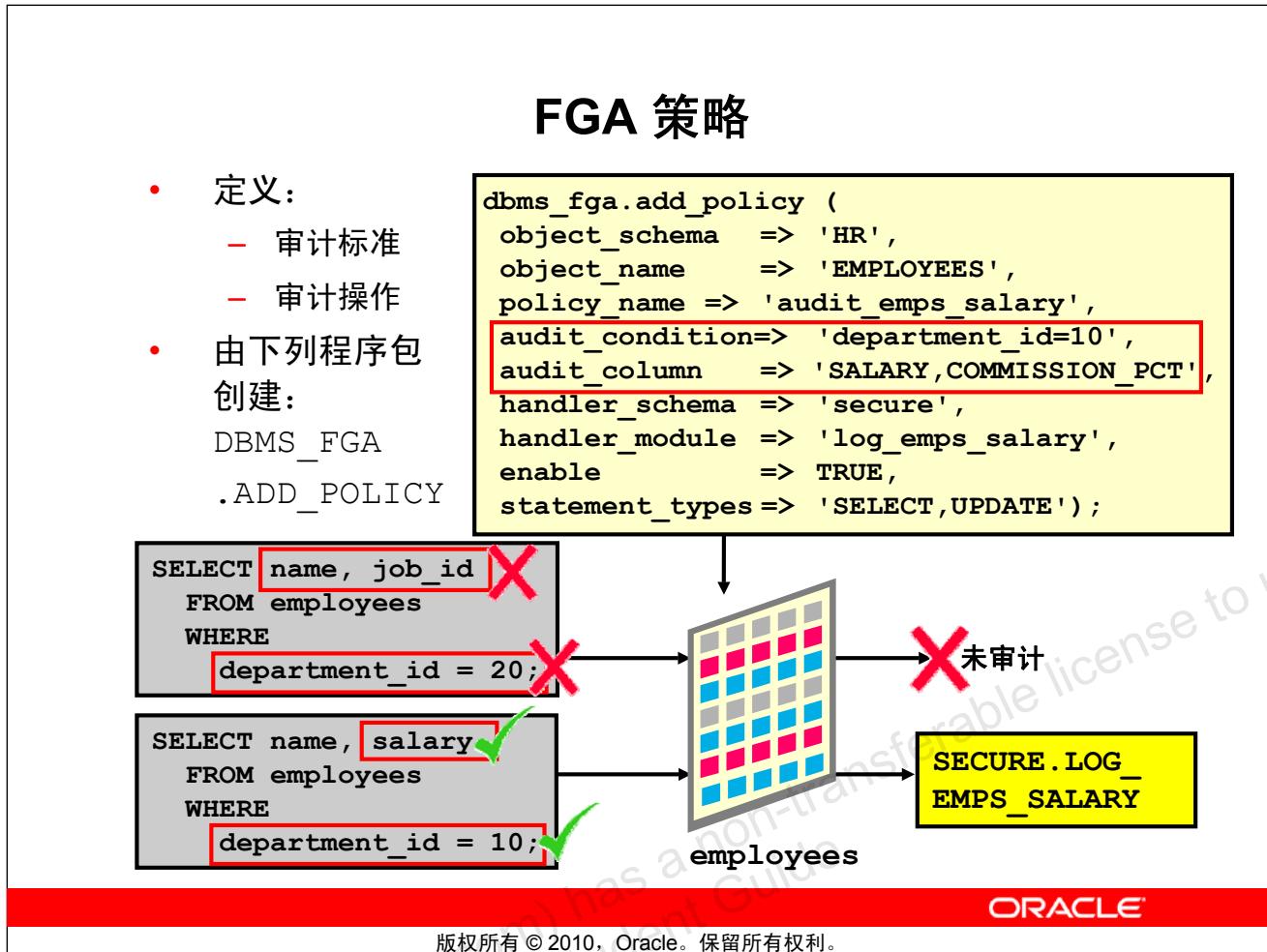
### 细粒度审计

数据库审计会记录已发生某个操作这一事实，但是不会捕获关于导致操作的语句的信息。细粒度审计(FGA)扩展了审计功能，可捕获查询或处理数据的实际SQL语句。

与标准数据库审计或基于值的数据库审计相比，FGA将重点审计的范围设置得更窄。

可将FGA选项设置为按表或视图中的各个列进行审计，还可将其设置为条件选项，以便只在符合管理员定义的特定规范时才捕获审计。FGA策略支持多个相关列。默认情况下，如果其中任何一列出现在SQL语句中，就会审计该语句。通过使用DBMS\_FGA.ALL\_COLUMNS和DBMS\_FGA.ANY\_COLUMNS，可对语句中是否使用了任何或全部相关列来进行审计。

可使用DBMS\_FGA PL/SQL程序包来创建对目标表或视图的审计策略。如果查询块中返回的任何行与审计列和指定的审计条件相匹配，则审计事件会导致在审计线索中创建并存储审计记录。此外，审计事件还可执行某个过程。FGA自动将审计重点放在语句级别。一个SELECT语句会返回数千行，但只生成一条审计记录。



## FGA 策略

幻灯片中的示例显示了使用 DBMS\_FGA.ADD\_POLICY 过程创建细粒度审计策略的流程，该过程接受以下参数。

### 策略名

创建每个 FGA 策略时您将指定其名称。幻灯片示例通过使用以下参数来命名

AUDIT\_EMPS\_SALARY 策略：

```
policy_name => 'audit_emps_salary'
```

### 审计条件

审计条件是一个 SQL 谓词，用于定义审计事件何时必须触发。在幻灯片示例中，通过使用以下条件参数来审计部门 10 中的所有行：

```
audit_condition => 'department_id = 10'
```

**注：** 细粒度审计查看查询的结果集，因此，使用幻灯片上显示的 FGA 策略时，返回与策略规范相匹配行的查询将导致创建一个审计记录。例如，查询“select \* from employees”会返回所有行，包括那些 department\_id 中含有“10”的行，因此将创建一个审计行。

## FGA 策略 (续)

### 审计列

审计列定义了要审计的数据。如果此列包括在 SELECT 语句中或是审计条件允许的选择，就会发生审计事件。幻灯片示例通过使用以下参数来审计两个列：

```
audit_column => 'SALARY,COMMISION_PCT'
```

此参数是可选参数。如果未指定此参数，则只由 AUDIT\_CONDITION 参数确定审计事件是否必须发生。

### 对象

对象是要审计的表或视图。可通过以下两个参数传递对象：

- 包含对象的方案
- 对象的名称

幻灯片示例通过使用以下参数来审计 hr.employees 表：

```
object_schema => 'hr'  
object_name => 'employees'
```

### 处理程序

可选的事件处理程序是一个 PL/SQL 过程，用于定义在审计期间必须执行的附加操作。例如，事件处理程序可向管理员发送一个预警页。如果未定义审计事件处理程序，则在审计线索中插入审计事件条目。如果定义了审计事件处理程序，则在审计线索中插入审计条目并执行审计事件处理程序。

审计事件条目包括导致事件的 FGA 策略、执行 SQL 语句的用户、SQL 语句及其绑定变量。

可通过以下两个参数传递事件处理程序：

- 包含 PL/SQL 程序单元的方案
- PL/SQL 程序单元的名称

幻灯片示例通过使用以下参数来执行 SECURE.LOG\_EMPS\_SALARY 过程：

```
handler_schema => 'secure'  
handler_module => 'log_emps_salary'
```

默认情况下，审计线索总是将 SQL 文本和 SQL 绑定信息写到 LOB。可以更改此默认设置（例如，系统遇到性能下降时）。

### 状态

状态指示是否启用了 FGA 策略。在幻灯片示例中，使用以下参数启用了此策略：

```
enable => TRUE
```

## 审计的 DML 语句：注意事项

- 如果满足 FGA 谓词并且引用了相关列，则会对记录进行审计。
- 不管指定列是什么，都会审计 DELETE 语句。
- 会审计 MERGE 语句以及生成的基础 INSERT、UPDATE 和 DELETE 语句。

未审计，因为没有涉及部门 10 的记录。

```
UPDATE hr.employees
SET salary = 1000
WHERE commission_pct = .2;
```

```
UPDATE hr.employees
SET salary = 1000
WHERE employee_id = 200;
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 审计的 DML 语句：注意事项

为 DML 语句定义了 FGA 策略时，如果要处理的数据行（包括新行和旧行）符合策略谓词标准，就会审计 DML 语句。

但是，如果同时在策略定义中指定了相关列，则当数据符合 FGA 策略谓词并且语句引用了定义的相关列时，才会审计语句。

对于 DELETE 语句，在定义策略期间指定相关列是没有用的，这是因为 DELETE 语句会访问表中的所有列。因此，不管相关列是什么，总是要审计 DELETE 语句。

FGA 支持 MERGE 语句。如果基础 INSERT、UPDATE 和 DELETE 语句满足所定义的 INSERT、UPDATE 或 DELETE FGA 策略，则会对这些语句进行审计。

使用前面定义的 FGA 策略时，会审计第一个语句而不审计第二个语句。部门 10 中没有雇员收到佣金，但是 employee\_id=200 指定了部门 10 中的一个雇员。

## FGA 准则

- 要审计所有行，请使用 null 审计条件。
- 要审计所有列，请使用 null 审计列。
- 策略名必须唯一。
- 创建策略时，审计的表或视图必须已经存在。
- 如果审计条件语法无效，则访问审计的对象时会出现 ORA-28112 错误。
- 如果表中不存在审计的列，则不会审计任何行。
- 如果不存在事件处理程序，并不会返回任何错误，仍会创建审计记录。



版权所有 © 2010, Oracle。保留所有权利。

### FGA 准则

对于 SELECT 语句，FGA 会捕获语句本身而不是实际行。但是，当 FGA 与闪回查询组合在一起时，会按实际行在该时间点存在的形式重新构造这些行。

有关闪回查询的详细信息，请参阅“执行闪回”一课。

有关 DBMS\_FGA 程序包的更多详细信息，请参阅《Oracle Database PL/SQL Packages and Types Reference》。

## SYSDBA 审计

具有 SYSDBA 或 SYSOPER 权限的用户可在数据库处于关闭状态时进行连接。

- 审计线索必须存储在数据库外部。
- 始终会对以 SYSDBA 或 SYSOPER 身份执行的连接进行审计。
- 可使用 AUDIT\_SYS\_OPERATIONS 启用对 SYSDBA 或 SYSOPER 操作的附加审计。
- 可使用 AUDIT\_FILE\_DEST 控制审计线索。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### SYSDBA 审计

SYSDBA 和 SYSOPER 用户有权启动和关闭数据库。因为这些用户可以在数据库处于关闭状态时进行更改，所以这些权限的审计线索必须存储在数据库外部。Oracle DB 会自动捕获 SYSDBA 和 SYSOPER 用户的登录事件。这为跟踪授权或未授权的 SYSDBA 和 SYSOPER 操作提供了一种宝贵方法，但只在查看操作系统审计线索时才有用。

Oracle DB 始终捕获授权用户的登录事件。如果专门启用了 DBA 审计，则还会捕获其它操作。通过设置以下初始化参数可启用对 SYSDBA 和 SYSOPER 用户的审计：

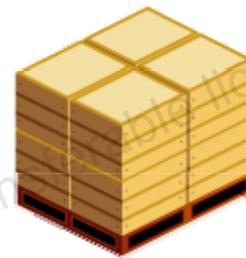
AUDIT\_SYS\_OPERATIONS=TRUE (默认设置为 FALSE。)

如果对 SYS 操作进行审计，则 AUDIT\_FILE\_DEST 初始化参数会控制审计记录的存储位置。在 Windows 平台上，默认情况下，审计线索存储在 Windows 事件日志中。在 UNIX 和 Linux 平台上，审计记录存储在 AUDIT\_FILE\_DEST 位置。

## 维护审计线索

应根据最佳实践准则来维护审计线索：

- 复查和存储旧记录。
- 避免出现存储问题。
- 避免记录丢失。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 维护审计线索

必须维护每种类型的审计线索。基本维护必须包括复查审计记录及删除数据库或操作系统中较旧的记录。审计线索会不断增长，可填满可用的存储空间。如果文件系统已满，系统可能会崩溃或只是导致出现性能问题。如果数据库审计线索填满了表空间，则不能完成审计的操作。如果审计线索填满了系统表空间，在审计操作停止之前，其它操作的性能会受到影响。

标准审计的审计线索存储在 AUD\$ 表中。FGA 的审计线索存储在 FGA\_LOG\$ 表中。默认情况下，这两个表都是在 SYSTEM 表空间中创建的。通过使用数据泵导出和导入实用程序可将这两个表移到另一表空间中。

**注：**不支持将审计表移到 SYSTEM 表空间外部。

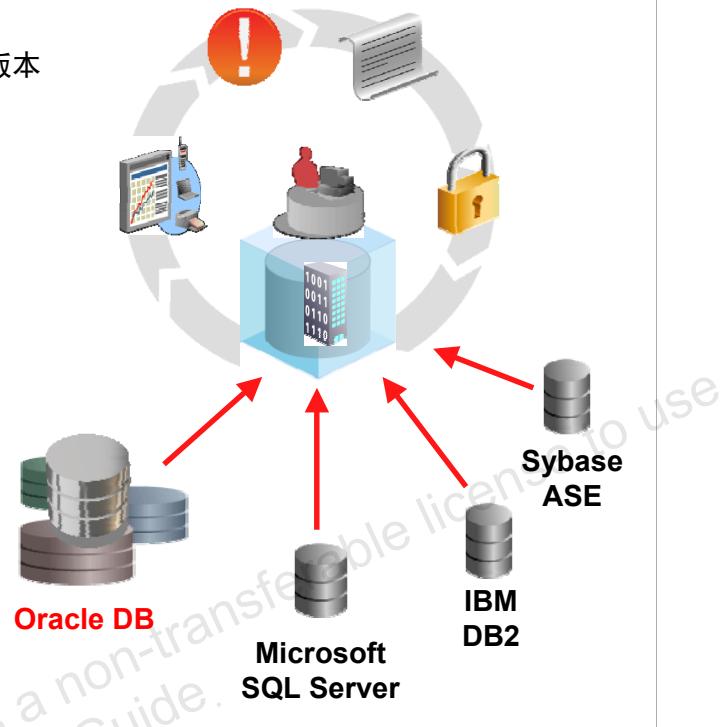
在从审计表中删除记录的过程中，可能会丢失审计记录。

#### 最佳实践提示

请根据时间戳执行导出，然后根据同一时间戳删除审计线索中的行。

## Oracle Audit Vault

- 合并和保护审计数据
  - Oracle 9i 发行版 2 和更高的版本
  - SQL Server 2000、2005
  - IBM DB2 UDB 8.5 和 9.2
  - Sybase ASE 12.5 - 15.0
  - 安全且可伸缩
  - 清理源 Oracle 审计数据
- 集中式报告
  - 报表界面进行了更新，使用广受欢迎的 Oracle Application Express
  - 合规性标准报表
  - 新增定制报表
- 安全威胁预警
  - 检测和警告安全相关事件



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle Audit Vault

Oracle Audit Vault 的主要优点如下：

- 从 Oracle9i Database 发行版 2 开始，Oracle Audit Vault 可以从 Oracle DB、Microsoft SQL Server 2000 和 2005、IBM DB2 Unix、Linux、Windows 8.2 和 9.5 以及 Sybase ASE 12.5 - 15.0 中透明地收集和合并审计数据。
- Oracle Audit Vault 提供内置报表和定制报表，帮助公司简化了制作合规性报表的工作。此外，Oracle Audit Vault 提供了可以从 Oracle BI Publisher、Oracle Application Express 或任何第三方报表工具访问的开放式审计仓库方案。
- Oracle Audit Vault 对可疑活动发出预警，从而帮助检测和防范内部威胁。
- Oracle Audit Vault 的核心是一个安全且可伸缩的审计仓库，它基于 Oracle 数据仓库技术构建并由 Oracle 的数据库安全产品（包括 Oracle Database Vault 和 Oracle 高级安全选件）保护安全。Oracle Audit Vault 包括 Oracle 分区功能，以提高易管理性和性能。
- Oracle Audit Vault 集中管理数据库审计设置（策略），可帮助公司降低 IT 成本，同时使 IT 安全负责人和内部审计人员能够更轻松地执行其工作。

有关更多信息，请参阅 Oracle Audit Vault 文档。

## 小测验

标准数据库审计捕获 DML 事务处理之前和之后的更改。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小测验

默认启用对 SYSDBA 和 SYSOPER 操作的审计。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小结

在本课中，您应该已经学会：

- 说明 DBA 负责的安全和审计工作
- 启用标准数据库审计
- 指定审计选项
- 复查审计信息
- 维护审计线索



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 11 概览：实施 Oracle DB 安全性

本练习包含以下主题：

- 启用标准数据库审计
- 指定 HR.JOBS 表的审计选项
- 更新表
- 复查审计信息
- 维护审计线索

ORACLE

版权所有 © 2010, Oracle。保留所有权利。



# 12

数据库维护

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

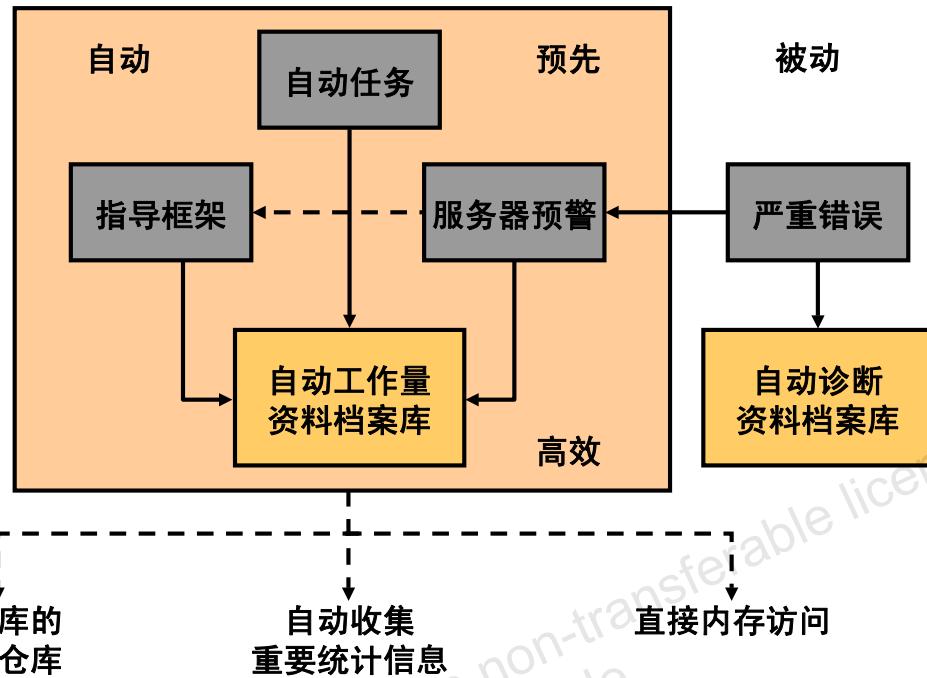
学完本课后，应能完成以下工作：

- 管理优化程序统计信息
- 管理自动工作量资料档案库 (AWR)
- 使用自动数据库诊断监视器 (ADDM)
- 说明和使用指导框架
- 设置预警阈值
- 使用服务器生成的预警
- 使用自动任务

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 数据库维护



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 数据库维护

通过复杂的 Oracle DB 基础结构，可以方便地执行预先数据库维护，主要包括以下要素：

- 每个 Oracle DB 中都有一个内置的资料档案库，即自动工作量资料档案库 (AWR)。Oracle DB 服务器会定期为所有重要统计信息及工作量信息创建快照，并将这些数据存储在 AWR 中。用户可以对捕获的数据进行分析，也可以由数据库自己进行分析，或者两者兼有。
- 数据库使用自动任务执行常规维护操作，如定期备份、刷新优化程序统计信息以及数据库健康检查。

被动数据库维护包括数据库健康检查器发现的严重错误和状态：

- 当出现无法自动解决并需要通知管理员的问题（如用完空间时）时，Oracle DB 服务器会提供服务器生成的预警。Oracle DB 服务器默认情况下会监视其本身，并发送预警通知您发生的问题。预警不仅通知您出现问题，通常还会就如何解决报告的问题提供建议。
- 建议由一些指导生成，其中每个指导负责一个子系统。例如，有内存指导、段指导和 SQL 指导。

## 查看预警历史记录

**Related Links**

Access	Add Exadata Cell Targets	Advisor Central
<b>Alert History</b>	Alert Log Contents	All Metrics
<a href="#">Baseline Metric Thresholds</a>	Blackouts	EM SQL History
Jobs	Metric and Policy Settings	Metric Collection Errors
Monitoring Configuration	Monitor in Memory Access Mode	Policy Groups
Scheduler Central	SQL Worksheet	Target Properties
User-Defined Metrics		

**Alert History**

Page Refreshed Jun 19, 2009 11:40:17 PM GMT+07:00  
View Data Last 24 hours

**Metric** History

Metric	History
Audited User	[Green]
<b>Instance Status</b>	[Red]
Mounted	[Green]

Key: Critical (Red), Warning (Yellow), Clear (Green), No Data (Grey)

11:40 3 6 9 12 PM 3 6 9  
18 June 2009

**Severity** **Timestamp** **Message** **Last Comment** **Details**

Severity	Timestamp	Message	Last Comment	Details
✓	Jun 19, 2009 2:02:12 AM	The instance is down, and health check reported: .	-	
✗	Jun 19, 2009 1:32:12 AM	The instance is down, and health check reported: Instance Shutdown.	-	
✓	Jun 18, 2009 5:31:15 AM	The instance is down, and health check reported: .	-	

ORACLE

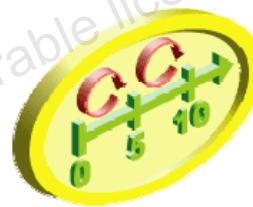
版权所有 © 2010, Oracle。保留所有权利。

## 查看预警历史记录

“Alert History（预警历史记录）”页会显示一个图表，图表上根据指定的时间段显示当前数据库的预警历史记录。出现预警表明可能存在问题：不是监视度量达到警告阈值或严重阈值，就是表明目标不再可用。单击“Alert History（预警历史记录）”页上列出的度量名称可获得每个预警的详细统计信息、图形和实际时间戳。此外，该页上还提供了一个输入预警相关注释（如解决方案信息）的位置。

## 术语

- 自动工作量资料档案库 (AWR): 是用于数据搜集、分析和提供解决方案建议的基础结构
- AWR 基线: 用于性能比较的一组 AWR 快照
- 度量: 累计统计数据中的更改率
- 统计信息: 提供数据库和对象详细信息的数据集合
  - 优化程序统计信息: 供查询优化程序使用
  - 数据库统计信息: 用于了解性能
- 阈值: 比较度量值时所根据的边界值



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 术语

自动工作量资料档案库 (AWR) 通过为内部 Oracle 服务器组件提供服务来收集、处理、维护和使用性能统计信息，以便进行问题检测和自优化。活动会话历史记录 (ASH) 是存储在 AWR 中的最近会话活动的历史记录。

统计信息是一些数据的集合，可提供有关数据库及数据库中对象的详细信息。查询优化程序会使用优化程序统计信息为每个 SQL 语句选择最佳执行计划。数据库统计信息则提供用于性能监视的信息。

AWR 快照包括数据库统计信息和度量、应用程序统计信息（事务处理量和响应时间）、操作系统统计信息及其它度量。AWR 基线是在一段时间内收集的一组 AWR 快照。基线用于性能比较，这既可以是当前性能与基线的比较，也可以是一个基线与另一基线的比较。

在 Oracle Database 11g 中，默认情况下会收集“系统移动窗口”基线。系统移动窗口基线是一组不断变化的快照，默认情况下包括最近八天的快照。收集了足够的数据并且计算了统计信息之后，此基线就会变为有效。默认情况下，统计信息计算被安排在每周六的午夜。

## Oracle 优化程序：概览

Oracle 优化程序可确定最有效的执行计划，这是处理任何 SQL 语句最重要的一步。

该优化程序可以：

- 对表达式和条件求值
- 使用对象和系统统计信息
- 确定如何访问数据
- 确定如何联接表
- 确定最有效的路径

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Oracle 优化程序：概览

优化程序是 Oracle DB 的一部分，用于为 SQL 语句创建执行计划。确定执行计划是处理任何 SQL 语句的重要一步，会对执行时间产生重大影响。

执行计划是执行语句时按顺序执行的一系列操作。优化程序会考虑许多与被引用对象相关的以及与查询中所指定的条件相关的因素。优化程序所需的信息包括：

- 为系统（I/O、CPU 等）以及方案对象（行数、索引等）搜集的统计信息
- 字典中的信息
- WHERE 子句限定符
- 开发人员提供的提示

使用诊断工具（如 Enterprise Manager、EXPLAIN PLAN 和 SQL\*Plus AUTOTRACE）时，可以看到优化程序选择的执行计划。

注：根据其功能的不同，Oracle 优化程序具有两个名称：“查询优化程序”和“自动优化程序”。

## 优化程序统计信息

优化程序统计信息：

- 是某一时间点的快照
- 每次重新启动实例后会变为永久信息
- 可自动收集

```
SQL> SELECT COUNT(*) FROM hr.employees;
      COUNT(*)
-----
      214
SQL> SELECT num_rows FROM dba_tables
  2 WHERE owner='HR' AND table_name = 'EMPLOYEES';
      NUM_ROWS
-----
      107
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 优化程序统计信息

优化程序统计信息包括表、列、索引和系统的统计信息。表和索引的统计信息存储在数据字典中。这些统计信息不提供实时数据。而是为优化程序提供数据存储和数据分发的正确统计快照，优化程序会使用该快照来决定如何访问数据。

收集的统计信息包括：

- 数据库块中表或索引的大小
- 行数
- 平均行大小和链计数（仅限表）
- 已删除叶行的高度和数量（仅限索引）

插入、删除和修改数据后，这些统计信息会发生变化。由于维护实时数据分发统计信息会对性能带来负面影响，因此，通过定期搜集表和索引的统计信息来更新这些统计信息。

优化程序统计信息由自动维护作业自动收集，该作业默认情况下在预定义的维护窗口中每日运行一次。系统统计信息是优化程序要使用的操作系统特征。不会自动收集这类统计信息。有关收集系统统计信息的详细信息，请参阅《Oracle Database Performance Tuning Guide》。

优化程序统计信息不同于 AWR 快照中搜集的数据库性能统计信息。

## 使用“管理优化程序统计信息”页

Database Instance: orcl.oracle.com

Home Performance Availability Server Schema Data Movement Software and Support

**Query Optimizer**

Manage Optimizer Statistics

SQL Plan Control

**Manage Optimizer Statistics**

Database orcl.oracle.com

Optimizer Statistics are used by the query optimizer to choose the best execution plan for each SQL statement. Up-to-date optimizer statistics can greatly improve the performance of SQL statements.

**Operations**

- [Gather Optimizer Statistics](#)
- [Restore Optimizer Statistics](#)
- [Lock Optimizer Statistics](#)
- [Unlock Optimizer Statistics](#)
- [Delete Optimizer Statistics](#)

**Related Links**

- [Object Statistics](#)
- [Global Statistics Gathering Options](#)
- [Object Level Statistics Gathering Preferences](#)
- [Job Scheduler](#)
- [Automated Maintenance Tasks](#)

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 使用“管理优化程序统计信息”页

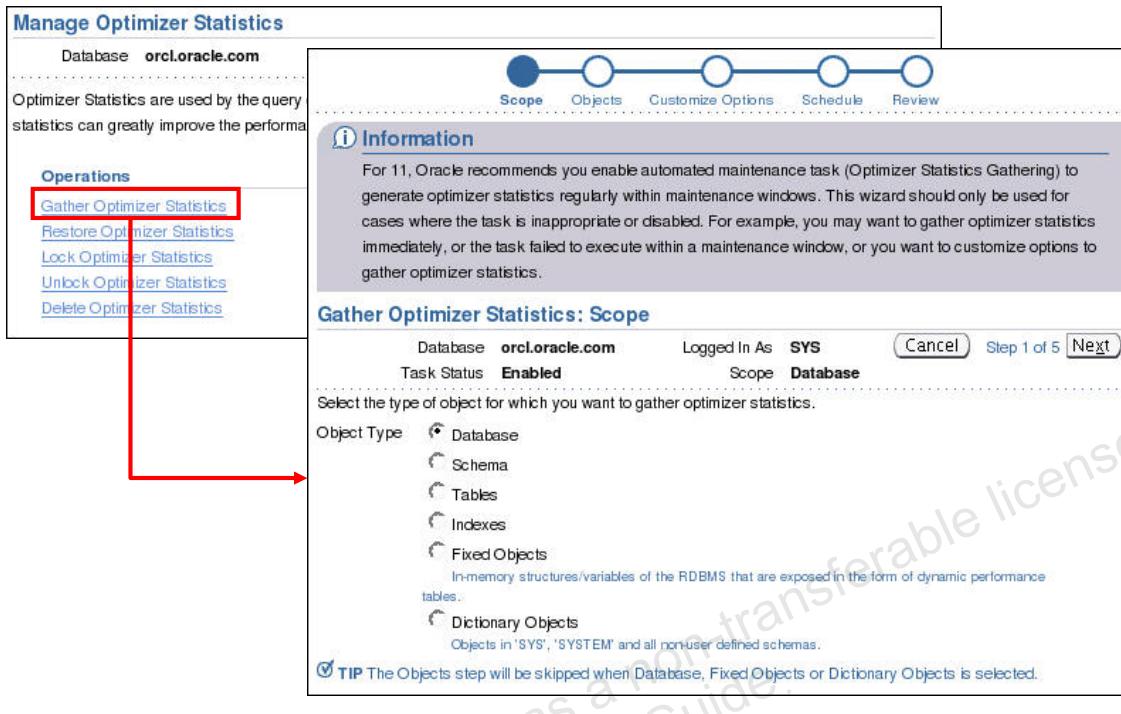
要在 Enterprise Manager 中管理优化程序统计信息，请单击“Server（服务器）”选项卡，然后单击“Query Optimizer（查询优化程序）”部分下的“Manage Optimizer Statistics（管理优化程序统计信息）”。在此页上，可以对统计信息执行下列任务：

- 手动搜集优化程序统计信息。
- 将优化程序统计信息还原到以前的某个时间点。选择的时间点必须位于优化程序统计信息保留期内，默认为 30 天。
- 锁定优化程序统计信息以确保永远不覆盖某些对象的统计信息。如果在存在很有代表性的数据时计算了某个表的统计信息，而您又希望永远保留这些统计信息，那么该任务就很有帮助。如果锁定了统计信息，表的任何波动都不会影响这些统计信息。
- 取消对优化程序统计信息的锁定以撤消以前执行的锁定。
- 删除优化程序统计信息以删除统计信息。

### 最佳实践提示

可使用自动维护任务搜集优化程序统计信息。要启用搜集优化程序统计信息的任务，必须确保 STATISTICS\_LEVEL 初始化参数设置为 TYPICAL 或 ALL。

# 手动搜集优化程序统计信息



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 手动搜集优化程序统计信息

某些时候可能需要手动搜集统计信息，例如表的内容在两次自动搜集作业之间发生了很大的变化，以致于统计信息无法再准确表示表。对于 24 小时内其大小变化超过 10% 的大型表，通常需要此操作。

**最佳实践提示：**收集统计信息的频率应足够高，以保证表在两个收集期之间的变化率不超过 10%。这可能需要手动收集统计信息或使用其它维护窗口。

可以使用 Enterprise Manager 或 DBMS\_STATS 程序包手动收集统计信息。系统统计信息只能使用 DBMS\_STATS 程序包来搜集。系统统计信息向查询优化程序描述了系统的硬件特征，例如 I/O 以及 CPU 性能和利用率。

选择“Gather Optimizer Statistics（搜集优化程序统计信息）”菜单会启动一个向导，使用该向导可以为要搜集优化程序统计信息的作业选择作用域、对象、选项和调度。该向导会在您指定的以下作用域提交 DBMS\_STATS.GATHER\_\*\_STATS 作业：表、方案或数据库。在该向导中，将首选项设置为 DBMS\_STATS 程序包所使用的默认值，并将此作业安排在您决定的时间运行。

不建议手动搜集常规统计信息，因为通过维护窗口搜集统计信息的效率更高，而且对用户的影响更小。如果自动作业已失败或已禁用，也可提交手动作业。

## 手动搜集优化程序统计信息（续）

此外，还可以直接通过 DBMS\_STATS 程序包搜集优化程序统计信息：

```
SQL> EXEC dbms_stats.gather_table_stats('HR', 'EMPLOYEES');
SQL> SELECT num_rows FROM dba_tables
  2 WHERE owner='HR' AND table_name = 'EMPLOYEES';
  NUM_ROWS
-----
      214
```

注意，现在的行数正确地反映了截止到搜集统计信息时表中的行数。通过 DBMS\_STATS 还可对整个方案甚至整个数据库启用手动收集统计信息。

除非工作量发生了显著变化，否则系统统计信息不会改变。因此，不需要频繁调整系统统计信息。DBMS\_STATS.GATHER\_SYSTEM\_STATS 过程会在指定时间段内收集系统统计信息，您也可以启动系统统计信息的搜集过程并执行另一个调用来停止搜集。

**最佳实践提示：**创建数据库时使用以下命令：

```
SQL> EXEC dbms_stats.gather_system_stats('NOWORKLOAD');
```

NOWORKLOAD 选项会花费几分钟的时间（具体取决于数据库大小）来捕获 I/O 特征的估计值，如平均读取查找时间和 I/O 传输率。

## 用于搜集统计信息的首选项



优化程序  
统计信息  
搜集任务

### 作用域

语句级  
表级  
方案级  
数据库级  
全局级

### 首选项

CASCADE  
DEGREE  
ESTIMATE\_PERCENT  
NO\_INVALIDATE  
METHOD\_OPT  
GRANULARITY  
INCREMENTAL  
PUBLISH  
STALE\_PERCENT



DBMS\_STATS

设置 | 获取 | 删除 | 导出 | 导入

```
exec dbms_stats.set_table_prefs('SH','SALES','STALE_PERCENT','13');
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 用于搜集统计信息的首选项

可在各种级别调用 DBMS\_STATS.GATHER\_\*\_STATS 过程，以便搜集整个数据库或单个对象（比如表）的统计信息。调用 GATHER\_\*\_STATS 过程时，通常可将其中的几个参数设置为默认值。提供的默认值适用于数据库中的多数对象，但对于某些对象或方案，需要更改这些默认值。在 Oracle Database 11g 中，可以为单个对象、方案或数据库设置值（称为“首选项”），也可以使用全局级命令更改默认值，而无需为每个对象运行手动作业。

这些首选项指定了赋予搜集过程的参数。SET\_\*\_PREFS 过程为不属于 SYS 或 SYSTEM 的任何对象创建首选项值。对于应位于数据库范围内的任何参数，一般由 DBA 为其设置全局首选项。这些过程适用于任何可设置为默认值的参数。

设置指定的首选项时，SET\_DATABASE\_PREFS 过程会循环访问数据库中的所有表和方案。SET\_SCHEMA\_PREFS 循环访问指定方案中的表。SET\_TABLE\_PREFS 为单个表设置首选项值。

无论是在数据库级、方案级还是在表级设置，所有对象首选项均保存在一个单个表中。在方案级更改这些首选项会覆盖以前在表级设置的首选项。

## 用于搜集统计信息的首选项（续）

在执行各种搜集过程时，这些过程会检索已经为每个对象设置的对象级首选项。您可以在 DBA\_TAB\_STAT\_PREFS 视图中查看这些对象级首选项。未在对象级设置的所有首选项都会被设置为全局级首选项。通过对每个首选项调用 DBMS\_STATS.GET\_PREFS 过程，可以查看全局首选项。

您可以在表级、方案级、数据库级和全局级设置、获取、删除、导出和导入这些首选项。首选项的值一般来说是按全局级向表级的顺序进行设置，最终将首选项应用于最小的组。

Oracle Database 11g 中的首选项：

- CASCADE，确定在搜集表统计信息的过程中是否收集索引统计信息。
- DEGREE，设置用于搜集统计信息的并行度。
- PUBLISH，用于确定是将统计信息发布到字典还是将其存储在专用区域中。这使得 DBA 可以先验证统计信息，然后再使用 PUBLISH\_PENDING\_STATS 过程将其发布到数据字典。
- STALE\_PERCENT 用于确定判断对象是否具有过时统计信息所依据的阈值级别。该值是自上次搜集统计信息以来已修改的行数百分比。示例仅将 SH.SALES 的默认值 10% 更改为 13%。
- INCREMENTAL，以增量方式搜集分区表的全局统计信息。
- METHOD\_OPT，确定列和用于搜集列统计信息的直方图参数。
- GRANULARITY，确定收集统计信息的粒度（仅当表为分区表时才适用）。
- NO\_INVALIDATE，确定是否使游标失效。
- ESTIMATE\_PERCENT，确定为获得有效的统计信息而采样的行数。它是表中行数的百分比。

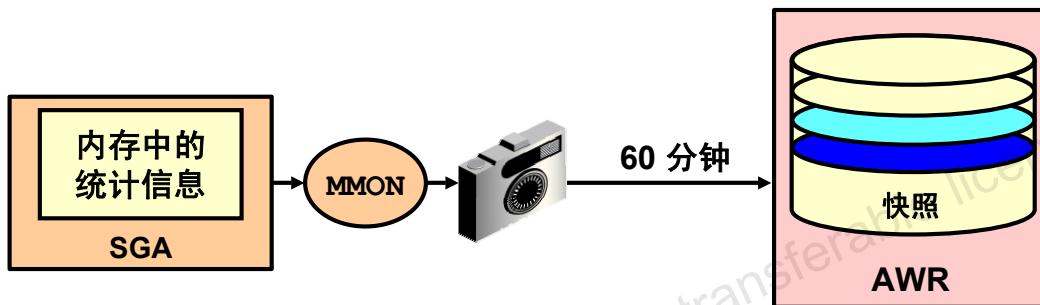
注：有关这些首选项的详细信息，请参阅《Oracle Database PL/SQL Packages and Types Reference》中的“DBMS\_STATS”文档。

使用 DBMS\_STATS.DELETE\_\*\_PREFS 过程，可在表级、方案级和数据库级删除首选项。

使用 DBMS\_STATS.RESET\_PARAM\_DEFAULTS 过程，可将全局首选项重置为建议值。

## 自动工作量资料档案库 (AWR)

- 内置的性能信息资料档案库
- 每 60 分钟获取一次数据库度量快照，保留期为八天
- 所有自我管理功能的基础



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

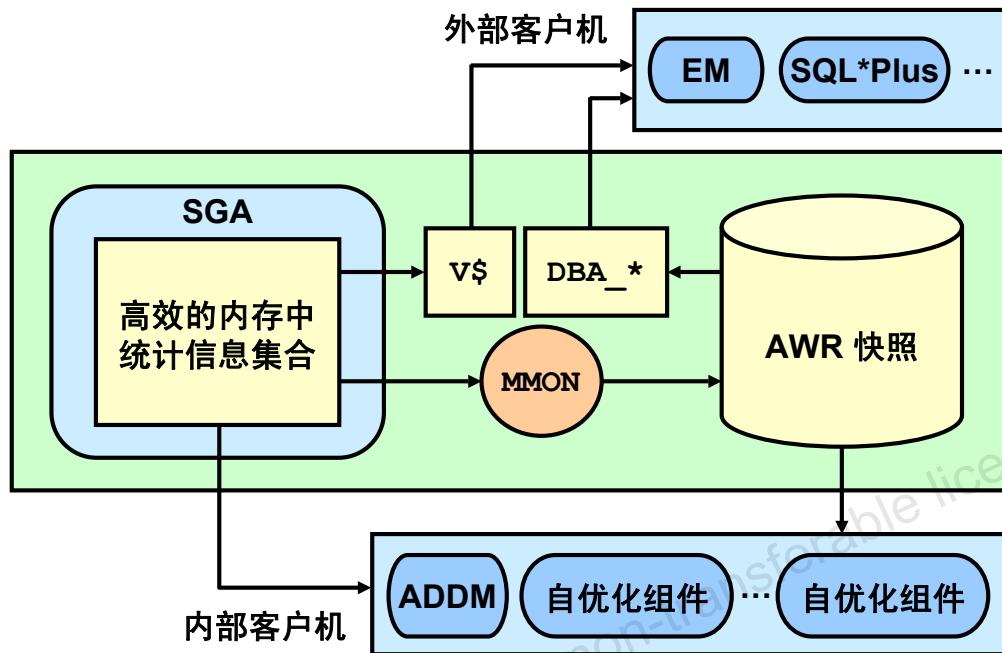
### 自动工作量资料档案库 (AWR)

AWR 是为 Oracle Database 11g 组件提供服务的基础结构，借助它可以收集、维护和利用统计信息进行问题检测和自优化。可将此基础结构视为数据库统计信息、度量等的数据仓库。

数据库（默认情况下）每 60 分钟从 SGA 中自动捕获一次统计信息，然后将其以快照形式存储在 AWR 中。这些快照通过一个名为易管理性监视器 (MMON) 的后台进程存储在磁盘上。默认情况下，快照会保留八天。您可以修改快照时间间隔和保留间隔。

AWR 中包含数百个表，所有这些表均属于 SYSMAN 方案且存储在 SYSAUX 表空间中。Oracle 建议仅使用 Enterprise Manager 或 DBMS\_WORKLOAD\_REPOSITORY 程序包访问资料档案库，对 AWR 进行操作。不支持直接对资料档案库表进行 DML 操作。

## AWR 基础结构



### AWR 基础结构

AWR 基础结构由两个主要部分组成:

- 一个内存中统计信息收集工具, Oracle Database 11g 组件使用它来收集统计信息。出于性能方面的考虑, 这些统计信息存储在内存中。可以通过动态性能 (V\$) 视图访问存储在内存中的统计信息。
- 代表该工具持久部分的 AWR 快照。AWR 快照可以通过数据字典视图和 Enterprise Manager Database Control 来访问。

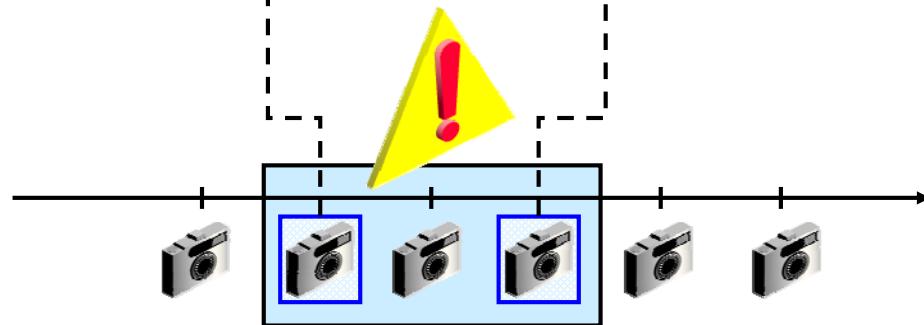
出于以下几方面的考虑, 统计信息存储在持久存储中:

- 实例崩溃后统计信息需要仍然可用。
- 某些分析需要使用历史记录数据进行基线比较。
- 可能会发生内存溢出。当旧统计信息因内存不足而被新统计信息替换时, 被替换的数据可以存储起来供以后使用。

内存版本的统计信息定期通过 MMON 后台进程转移到磁盘上。使用 AWR 时, Oracle DB 可自动捕获历史统计信息, 而不需要 DBA 进行干预。

## AWR 基线

→ 过去的相关时段 ←



```
DBMS_WORKLOAD_REPOSITORY.CREATE_BASELINE ( -
    start_snap_id IN NUMBER,
    end_snap_id   IN NUMBER,
    baseline_name IN VARCHAR2);
```

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### AWR 基线

AWR 基线是一个 AWR 快照集。通常是在 AWR 中标记和保留的某个重要时段的快照数据的集合。基线是用一对快照定义的；这两个快照通过其快照序列号 (snap\_id) 或起始和结束时间进行标识。每个快照集都有起始和结束快照，并包含位于这两个快照之间的所有快照。快照集用于保留快照数据。因此，默认情况下，在删除快照集前，属于快照集的快照会一直保留。可以根据快照要保留的天数设置一个过期值。

基线通过用户提供的名称来标识。执行 CREATE\_BASELINE 过程可基于一个快照集创建基线，并可以指定名称和一对快照标识符。对于新创建的基线，会为其分配一个在数据库生命周期内唯一的基线标识符。通常是基于过去有代表性的时段建立快照集，以用于与当前系统行为进行比较。您也可以在 Database Control 中使用基线来设置基于阈值的预警。可以使用该过程的过期参数设置以天数表示的过期时间。默认值为 NULL，表示“永不过期”。

可从 DBA\_HIST\_SNAPSHOT 或 Database Control 中直接获取 snap\_id。

**注：**有关 DBMS\_WORKLOAD\_REPOSITORY 程序包的详细信息，请参阅《Oracle Database PL/SQL Packages and Types Reference》指南。

## Enterprise Manager 和 AWR

The screenshot shows the Oracle Enterprise Manager interface. At the top, there is a navigation bar with tabs: Home, Performance, Availability, Server (which is selected), Schema, and Data Movement. On the left, there is a sidebar with sections for Storage (Control Files, Tablespaces, Temporary Tablespace Groups, Datafiles, Rollback Segments, Redo Log Groups, Archive Logs, Migrate to ASM, Make Tablespace Locally Managed) and Statistics Management (Automatic Workload Repository, AWR Baselines). The main content area is titled 'Automatic Workload Repository' and displays the following information:

- Page Refreshed Aug 21, 2008 9:47:11 PM MDT (with a Refresh button)
- The Automatic Workload Repository is used for storing database statistics that are used for performance tuning.
- General** section:
  - Snapshot Retention (days) 8
  - Snapshot Interval (minutes) 60
  - Collection Level TYPICAL
  - Next Snapshot Capture Time Aug 21, 2008 10:00:49 PM
- Manage Snapshots and Baselines** section:
  - Run AWR Report
  - Snapshots 195
  - Baselines 1
  - Latest Snapshot Time Aug 21, 2008 9:00:49 PM
  - Earliest Snapshot Time Aug 13, 2008 7:00:26 PM

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### Enterprise Manager 和 AWR

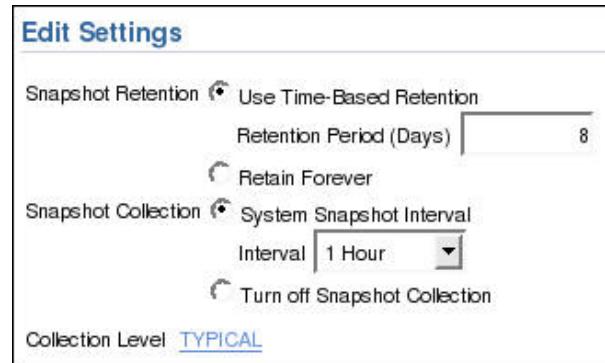
单击“Server（服务器）”选项卡，然后单击“Statistics Management（统计信息管理）”部分中的“Automatic Workload Repository（自动工作量资料档案库）”。在“Automatic Workload Repository（自动工作量资料档案库）”页上，单击“Edit（编辑）”更改设置。

在“Automatic Workload Repository（自动工作量资料档案库）”页上，可执行下列操作：

- 编辑工作量资料档案库设置
- 查看已创建的快照的详细信息，或者手动创建新快照
- 创建 AWR 基线
- 生成 AWR 报表

## 管理 AWR

- 保留期
  - 默认: 八天
  - 考虑存储需要
- 收集间隔
  - 默认: 60 分钟
  - 考虑存储需要和性能影响
- 收集级别
  - Basic (禁用多数 ADDM 功能)
  - Typical (建议级别)
  - All (将其它 SQL 优化信息添加到快照中)



**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

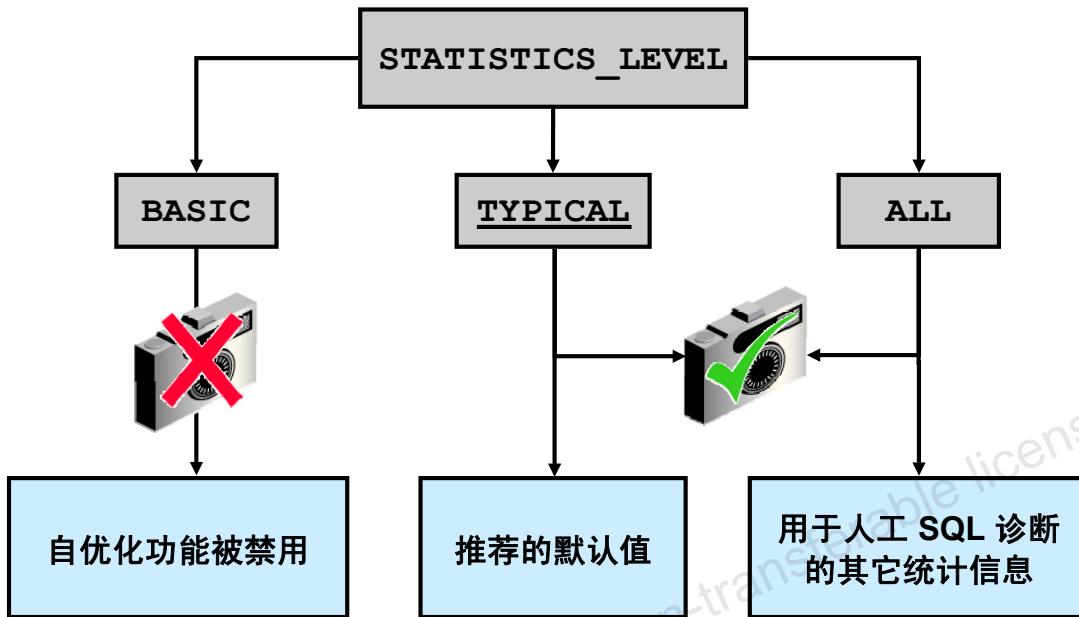
### 管理 AWR

AWR 设置其中包括保留期、收集间隔和收集级别。请记住，降低其中任何设置都会影响依赖于 AWR 的组件（包括指导）的功能。

增大设置可改进指导建议，但是会增加存储快照所需的空间，而且收集快照信息时的性能会下降。

优化新应用程序时，请考虑将收集级别设置为 ALL。使用 ALL 设置可收集 SQL 执行计划和定时统计信息，这些信息可以改进 SQL 指导的建议。优化完成后，应重新将此设置设回 TYPICAL。

## 统计级别



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 统计级别

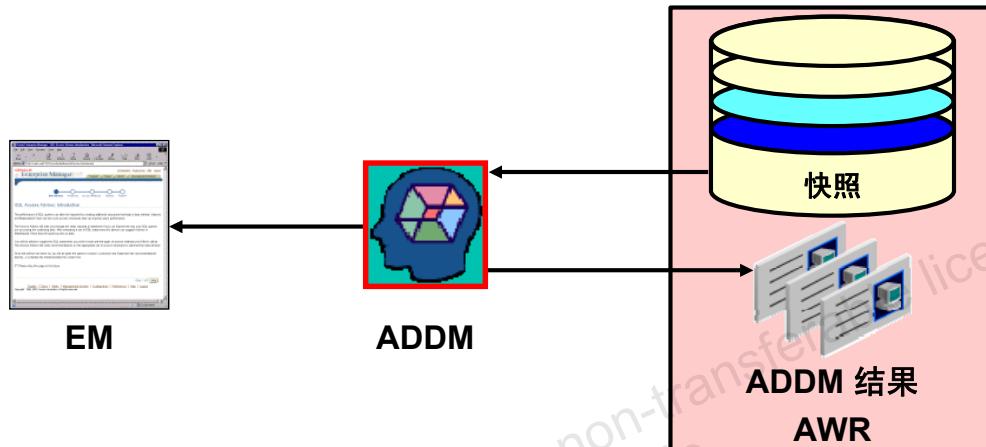
STATISTICS\_LEVEL 初始化参数可控制对各种统计信息和各种指导的捕获，其中包括自动维护任务。自动维护任务包括搜集优化程序统计信息。STATISTICS\_LEVEL 参数可设置为以下级别：

- **BASIC:** AWR 统计信息和度量的计算被关闭。自动优化程序统计信息任务被禁用，所有指导和服务器生成的预警也被禁用。
- **TYPICAL:** 收集数据库自我管理所需的主要统计信息。这些统计信息表示监视 Oracle DB 行为通常需要的信息。这包括自动搜集统计信息，以减少由于统计信息过时或无效而导致不正确执行 SQL 语句的可能性。
- **ALL:** 捕获所有可能的统计信息。此捕获级别增加了计时操作系统统计信息和计划执行统计信息。多数情况下并不需要这些统计信息，为了获得最佳性能，不应启用这些统计信息；有时执行特定的诊断测试时需要这些统计信息。

Oracle 建议将 STATISTICS\_LEVEL 初始化参数设置为默认值 TYPICAL。如果将该参数的值设置为 BASIC，则会禁用自动搜集优化程序统计信息功能。

## 自动数据库诊断监视器 (ADDM)

- 在记录每个 AWR 快照之后运行
- 监视实例；检测瓶颈
- 在 AWR 中存储结果



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 自动数据库诊断监视器 (ADDM)

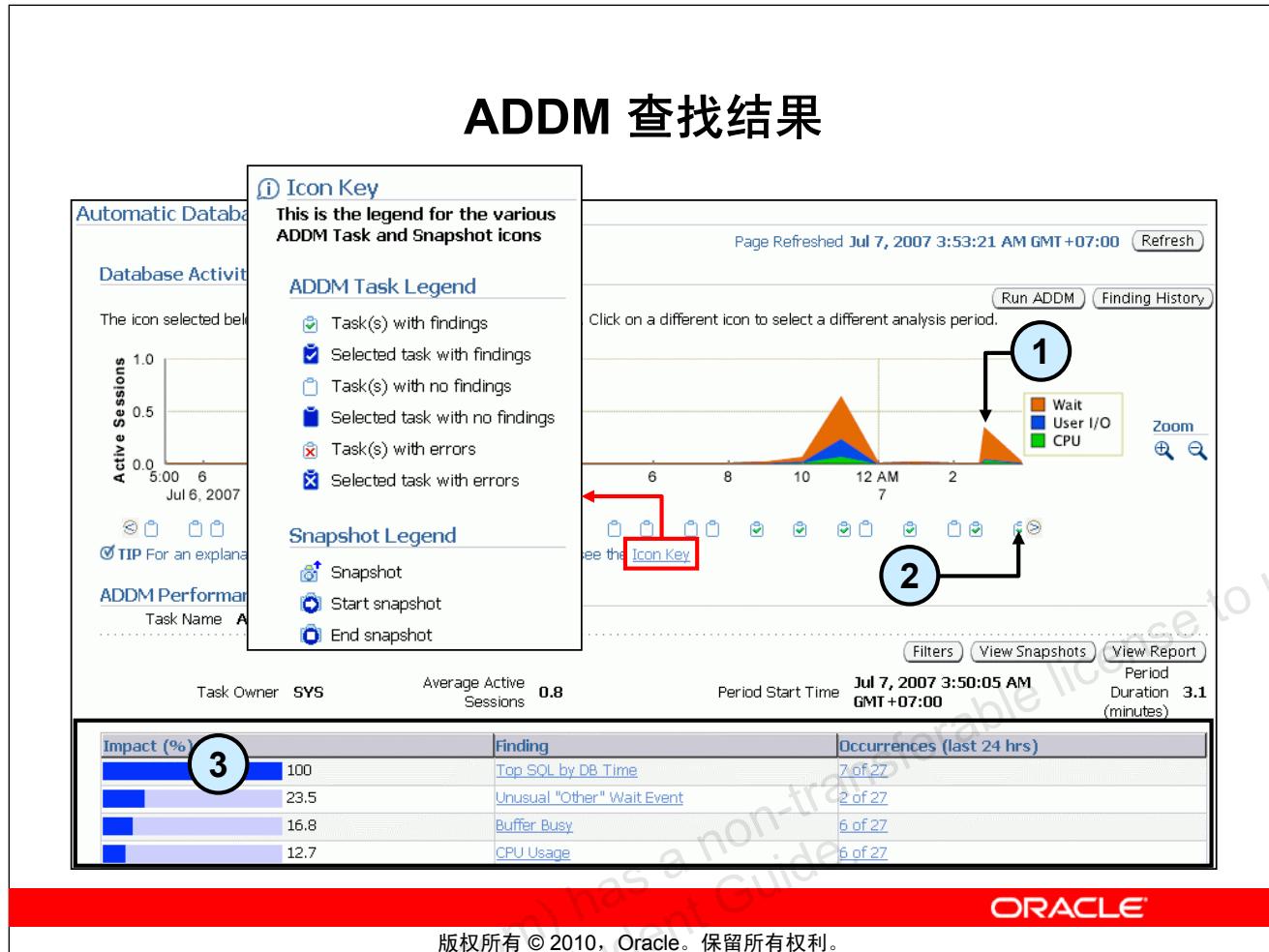
与其它指导不同，ADDM 在记录每个 AWR 快照之后会自动运行。每次记录快照后，ADDM 会分析与最后两个快照对应的时段。ADDM 会预先监视实例，以便在大多数瓶颈问题成为严重问题之前检测到这些问题。

多数情况下，ADDM 会为检测到的问题提供建议解决方案，甚至可以量化这些建议的优势。

ADDM 检测到的一些常见问题包括：

- CPU 瓶颈
- Oracle Net 连接管理不佳
- 锁争用
- 输入/输出 (I/O) 能力差
- 数据库实例内存结构大小不足
- SQL 语句的负载过高
- PL/SQL 和 Java 时间过高
- 检查点负载过高及原因（例如，日志文件太小）

每次 ADDM 分析的结果都存储在 AWR 中，并可通过 Enterprise Manager 进行访问。



## ADDM 查找结果

在“Automatic Database Diagnostic Monitor (ADDM)（自动数据库诊断监视器(ADDM)）”页中，可查看上次运行 ADDM 的详细查找结果。数据库时间表示会话在分析时段在数据库中花费的非空闲时间总和。为每个查找结果都提供了一个特定的影响百分比。此影响百分比表示相应问题消耗的时间与分析时段的数据库时间之比。

在幻灯片中，请注意以下几点：

1. 本图说明此时平均活动用户数显著增加。另外，主要问题是 Wait 问题。
2. 此图标表示在此页底部显示的 ADDM 输出对应于此时间点。单击其它图标可转到过去的时间（用于查看以前的分析）。
3. 这些查找结果提供一个简短概要，按可优化区域显示 ADDM 找到的问题。单击特定问题，可以转至“Performance Finding Details（性能查找结果详细资料）”页。

单击“View Report（查看报表）”按钮可以获得文本报表格式的性能分析详细资料。

# ADDM 建议

**Performance Finding Details: Buffer Busy**

Finding Read and write contention on database blocks was consuming significant database time. [Finding History](#)

Impact (Active Sessions) .14  
Impact (%) 16.8  
Period Start Time Jul 7, 2007 3:50:05 AM GMT+07:00  
Period Duration (minutes) 3.1  
Filtered No [Filters](#)

**Recommendations**

Show All Details | Hide All Details

Details	Category	Benefit (%) ▾
▼ Hide	Schema	16.8
Action	Consider using ORACLE's recommended solution of automatic segment space management in a locally managed tablespace for the tablespace "TBSSPC" containing the TABLE "SPC.SPCT" with object ID 82664. Alternatively, you can move this object to a different tablespace that is locally managed with automatic segment space management.	
Database Object	SPC.SPCT	
► Show	Schema	16.8
► Show	Schema	16.8

**Rationale** There was significant read and write contention on TABLE "SPC.SPCT" with object ID 82664.  
Database Object SPC.SPCT

► Show	Schema	16.8
► Show	Schema	16.8

**Findings Path**

Expand All | Collapse All

Findings	Impact (%)	Additional Information
▼ Read and write contention on database blocks was consuming significant database time.	16.8	
Wait class "Concurrency" was consuming significant database time.	17.4	

ORACLE

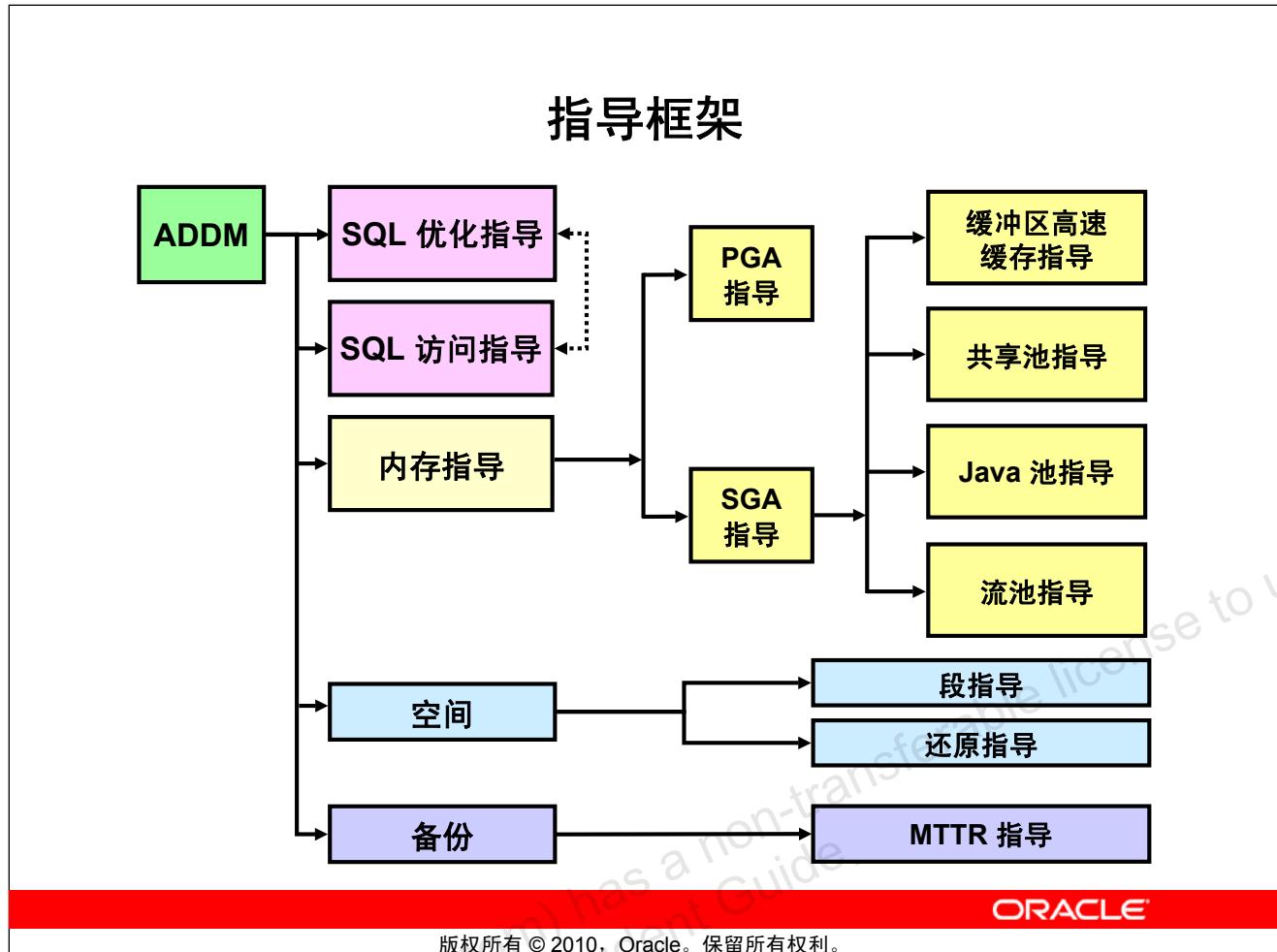
版权所有 © 2010, Oracle。保留所有权利。

## ADDM 建议

在“Performance Finding Details（性能查找结果详细资料）”页中，会得到用于解决相应问题的建议。这些建议分为“Schema（方案）”、“SQL Tuning（SQL 优化）”、“Database Configuration（数据库配置）”和其它许多类别。如果实施建议，则“Benefit（%）（缩时比率（%））”列将显示数据库已用时间的最大减少量。

ADDM 将考虑对系统的各种更改。其建议可以包括：

- **硬件更改：**添加 CPU 或更改 I/O 子系统配置
- **数据库配置：**更改初始化参数设置
- **方案更改：**对表或索引进行散列分区，或者使用自动段空间管理 (ASSM)
- **应用程序更改：**对序列使用高速缓存选项，或使用绑定变量
- **使用其它指导：**在高负载 SQL 上运行 SQL 优化指导或在热对象上运行段指导



## 指导框架

指导可提供有关资源占用率和各个服务器组件性能的有用反馈。例如，内存指导为 MEMORY\_TARGET 初始化参数提供建议值，该参数用于控制 Oracle DB 实例所使用的总内存量。

由于 ADDM 依赖于 AWR 中捕获的数据，因此 Oracle DB 通过 ADDM 可诊断自身的性能并确定如何解决识别出的问题。ADDM 在每次捕获 AWR 统计信息后会自动运行。它可能还会调用其它指导。

下面是指导基础结构提供的主要优点：

- 所有指导都使用统一的界面。
- 通过使用工作量资料档案库，所有指导拥有共用的数据源和结果存储空间。

幻灯片中未显示所有指导（例如，未列出数据恢复指导和 SQL 修复指导）。

### 自动数据库诊断监视器 (ADDM)

ADDM 是一个基于服务器的专用软件，它每隔 60 分钟复查一次数据库的性能。ADDM 的目标是提前检测出可能存在的系统瓶颈，并在系统性能明显降低之前提供建议的修复办法。

## 指导框架（续）

### 内存指导

内存指导实际上是多项指导功能的集合，通过它可确定数据库实例所使用的总内存的最佳设置。系统全局区 (SGA) 为共享池、数据库缓冲区高速缓存、Java 池和流池提供了一组指导。Java 池和流池指导不显示在 EM 的“Memory Advisor（内存指导）”页上。还有一个指导针对程序全局区 (PGA)。除了指导功能外，本指导还为大型池和 Java 池提供了一个中央控制点。

### 平均恢复时间 (MTTR) 指导

使用 MTTR 指导，可设置实例崩溃后数据库恢复所需的时间长短。

### 段指导

此指导用于查找占用空间多于所需空间的表和索引。此指导会在表空间级或方案级检查造成低效率的空间消耗问题，如果可能，还会生成脚本来减少空间消耗。

### SQL 访问指导

此指导用于分析在给定时段发出的所有 SQL 语句，还就其它索引或实体化视图（可提高性能）的创建提供建议。

### SQL 优化指导

此指导用于分析单个 SQL 语句，还提供建议以改进该语句的性能。建议中可包括重写语句、更改实例配置或添加索引等操作。不会直接调用 SQL 优化指导，而是从其它工具（如顶级 SQL 或顶级会话）中调用它，以帮助优化影响大的 SQL 语句。

### 还原管理指导

使用还原管理指导时，可确定支持指定的保留期所需要的还原表空间大小。“管理还原数据”一课中介绍了还原管理及还原指导的用法。

### 数据恢复指导

此指导自动诊断持续性数据故障，向用户提供修复选项并根据用户的请求执行修复。数据恢复指导的用途是减少平均恢复时间 (MTTR) 并提供用于自动修复数据的集中式工具。

### SQL 修复指导

如果某一 SQL 语句因严重错误而失败，进而导致在自动诊断资料档案库中生成问题，则可运行 SQL 修复指导。该指导会对该语句进行分析，并在多数情况下会推荐一个补丁程序来修复该语句。如果实施了建议，所应用的 SQL 补丁程序会让查询优化程序选择一个替代执行计划以备将来执行，从而避免故障。此操作无需更改该 SQL 语句本身即可完成。

# Enterprise Manager 和指导

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## Enterprise Manager 和指导

“Advisor Central（指导中心）”页是所有指导的主页。在 Database Control 主页的“Related Links（相关链接）”列表中，单击“Advisor Central（指导中心）”链接可访问此页。但是，Database Control 中并不是只有这一个可调用指导的地方。还可以在某些上下文中访问指导。

在“Advisor Central（指导中心）”页的“Advisors（指导）”选项卡上，可以列出工作量资料档案库中注册的所有指导任务。您还可以按指导类型和预定义时段过滤此列表。

通过“Advisor Central（指导中心）”页的“Checkers（检查器）”选项卡，可以调度各种数据库完整性检查器。可按名称、类型或时段列出所有检查器运行。

某些指导将在“管理还原数据”、“性能管理”和“备份和恢复的概念”这些课中进行更详细地介绍。

**注：**使用“Change Default Parameters（更改默认参数）”页可更改将来所有任务的默认到期时间（以天为单位）。还可以使用此页更改某些重要指导的参数。

## DBMS ADVISED 程序包

过程	说明
CREATE_TASK	在资料档案库中创建新任务
DELETE_TASK	在资料档案库中删除任务
EXECUTE_TASK	开始执行任务
INTERRUPT_TASK	挂起当前正在执行的任务
GET_TASK_REPORT	创建指定任务的文本报表并返回文本报表
RESUME_TASK	恢复挂起的任务
UPDATE_TASK_ATTRIBUTES	更新任务属性
SET_TASK_PARAMETER	修改任务参数
MARK_RECOMMENDATION	将一个或多个建议标记为已接受、已拒绝或已忽略
GET_TASK_SCRIPT	创建由已接受的所有建议组成脚本

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

### DBMS ADVISED 程序包

DBMS ADVISED 程序包中包含所有指导模块的所有常数和过程声明。使用这个程序包可从命令行执行任务。

要执行指导过程，必须具有 ADVISOR 权限。使用 ADVISOR 权限可对指导过程和视图进行全权访问。

**注：**有关 DBMS ADVISED 程序包中的所有过程的详细信息，请参阅《Oracle Database PL/SQL Packages and Types Reference》。

## 小测验

优化程序统计信息 `num_rows` 总能正确反映表的行数。

- 1. 正确**
- 2. 错误**

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 自动维护任务

### 自动任务维护进程:

1. 打开维护窗口。
2. 自动任务后台进程调度作业。
3. 调度程序启动作业。
4. 资源管理器限制自动任务作业的影响。

### 默认自动任务维护作业:

- 搜集优化程序统计信息
- 自动段指导
- 自动 SQL 指导



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 自动维护任务

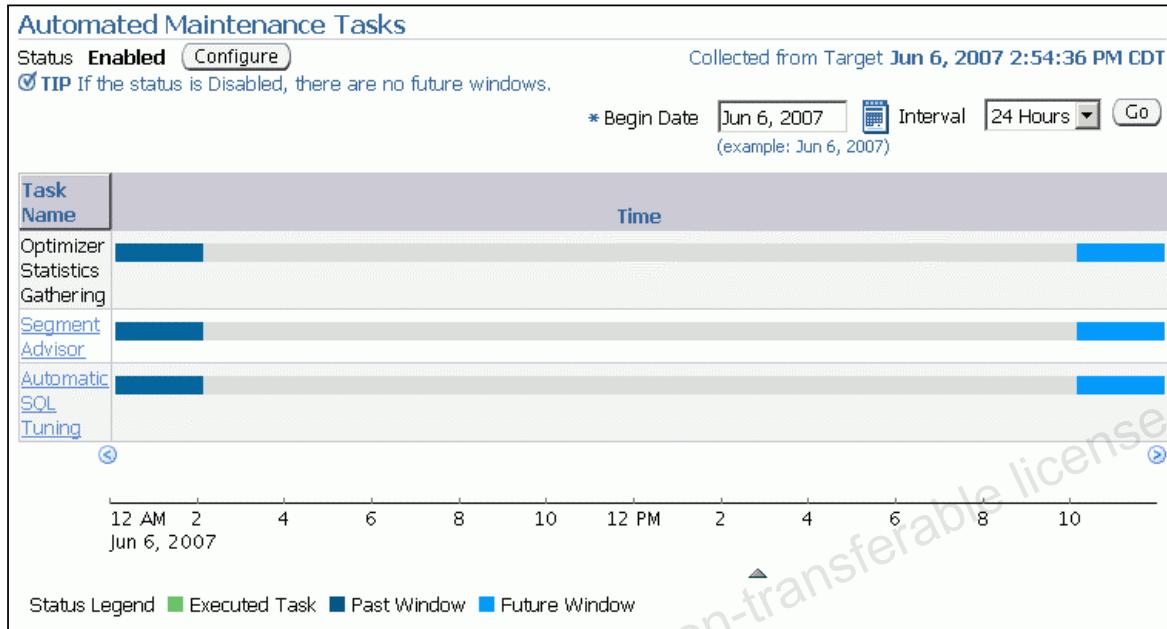
通过分析 AWR 中存储的信息，数据库可确定是否需要执行常规维护任务，如刷新优化程序统计信息。自动维护任务基础结构使 Oracle DB 能够自动执行这些操作。这种基础结构使用调度程序在预定义的维护窗口中运行这些任务。

默认情况下，工作日维护窗口在晚上 10 点启动，并持续 4 小时。在周六和周日，维护窗口在早上 6 点启动，并持续 20 小时。维护窗口的所有属性均可自定义，这些属性包括起始时间和结束时间、频率、每周几天等等。另外，通过将数据库资源管理器资源计划与维护窗口关联，可限制自动维护任务对正常数据库操作的影响。

#### 维护示例:

- 使用自动维护任务基础结构自动刷新优化程序统计信息。
- 自动段指导有默认的作业，这些作业在维护窗口中运行。
- 使用 DBCA 创建数据库时，可启动常规数据库备份。

## 自动维护任务



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 自动维护任务（续）

在“Server（服务器）”页上单击“Scheduler（调度程序）”标题下的“Automated Maintenance Tasks（自动维护任务）”可访问“Automated Maintenance Task（自动维护任务）”页，在该页可以查看自动维护任务调度和最近历史记录。从此处可细化到某些任务的详细资料。单击“Configure（配置）”进入“Automated Maintenance Tasks Configuration（自动维护任务配置）”页。任务将在窗口中执行。图中显示的是已在其中执行一个任务的上一个窗口和计划在其中执行该任务的下一个窗口。

**注：**示例中显示的是默认任务窗口。如果维护窗口关闭，则调度程序默认情况下会终止优化程序统计信息搜集作业。之后会在下一个维护窗口中处理剩余对象。

## 自动维护任务配置

**Automated Maintenance Tasks Configuration**

Global Status  Enabled  Disabled

**Task Settings**

Optimizer Statistics Gathering  Enabled  Disabled

Segment Advisor  Enabled  Disabled

Automatic SQL Tuning  Enabled  Disabled

**Maintenance Window Group Assignment**

Window	Optimizer Statistics Gathering	Segment Advisor	Automatic SQL Tuning
	<input type="button" value="Select All"/> <input type="button" value="Select None"/>	<input type="button" value="Select All"/> <input type="button" value="Select None"/>	<input type="button" value="Select All"/> <input type="button" value="Select None"/>
WEDNESDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
THURSDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
FRIDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
SATURDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
SUNDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
MONDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TUESDAY WINDOW	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

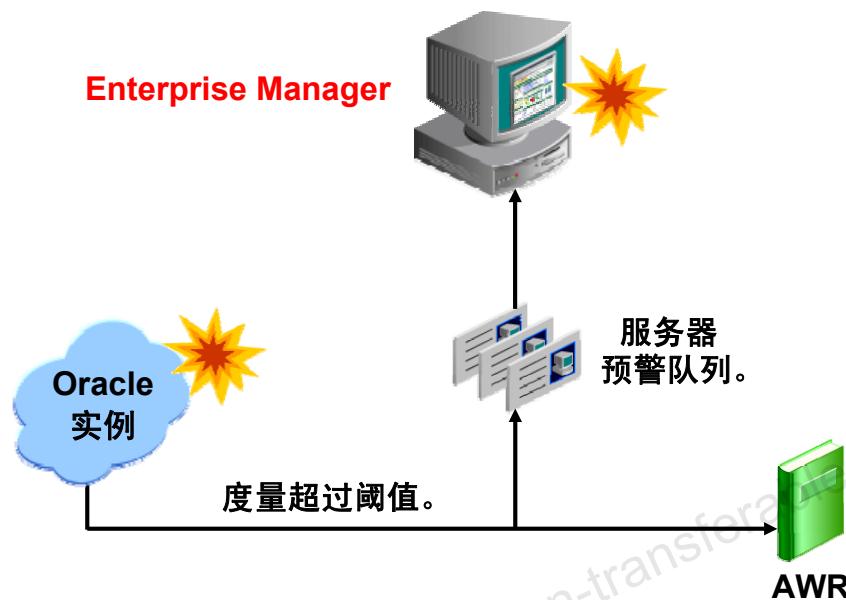
## 自动维护任务配置

在“Automated Maintenance Tasks Configuration（自动维护任务配置）”页上，可一次全部或者按单个任务或特定窗口启用和禁用自动维护任务。还可以配置用于搜集优化程序统计信息的设置以及自动 SQL 优化指导的作业控制参数。

选择窗口名称以查看或编辑窗口调度。

单击“Edit Window Group（编辑窗口组）”在窗口组中添加和删除窗口。

## 服务器生成的预警



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 服务器生成的预警

预警是当数据库处于不理想状态且需要引起注意时发出的通知。默认情况下，Oracle DB 通过 Enterprise Manager Database Control 提供预警。也可选择配置 Enterprise Manager 以向管理员发送有关问题状况的电子邮件，还可在控制台上显示预警信息。

另外，还可为系统的多个相关度量设置阈值。如果数据库偏离正常读数太多而达到了这些阈值，Oracle Database 11g 会预先发送通知。提前通知潜在的问题，可使用户迅速作出响应，并通常在用户注意到这些问题之前就解决这些问题。

默认情况下受监视的度量大约有 60 个，其中包括：

- 中断作业计数
- 等待所用的数据库时间百分比 (%)
- 转储区占用率 (%)
- 参照基线的 SQL 响应时间百分比 (%)
- 表空间占用率 (%)
- 一般意外事件

## 服务器生成的预警（续）

另外一些可提前提供问题通知的重要度量有：

- 平均文件读取时间（厘秒）
- 响应时间（每事务处理）
- 等待时间 (%)

## 设置阈值

Metric	Comparison Operator	Warning Threshold	Critical Threshold	Corrective Actions	Collection Schedule	Edit
Access Violation	Matches		*	None	Every 5 Minutes	
Access Violation Status	>		0	None	Every 5 Minutes	
Archive Area Used (%)	>	80		None	Every 15 Minutes	
Archiver Hung	Matches		*	None	Every 5 Minutes	
Archiver Hung Status	>		0	None	Every 5 Minutes	
Audited User	=	SYS		None	Every 15 Minutes	
Average Users Waiting Count						
Administrative	>	10		None		
Application	>	10		None		
Cluster	>	30		None		
Commit	>	30		None		
Concurrency	>	10		None		
Configuration	>	10		None		
Network	>	10		None		

**ORACLE**

版权所有 © 2010, Oracle。保留所有权利。

## 设置阈值

要设置或编辑整个数据库的阈值，请在数据库主页的“Related Links（相关链接）”区域中单击“Metric and Policy Settings（度量和策略设置）”。输入所需的警告阈值和严重阈值。数据库达到指定的值时，就会出现相应的预警。

已设置的阈值显示在“Metrics with thresholds（带有阈值的度量）”列表中。默认情况下，大约有 60 个度量包含预先设置的阈值；您可以根据需要更改这些阈值。“All metrics（所有度量）”列表会显示未设置阈值的度量。

单击其中一个“Edit（编辑）”图标可访问一个页面，在该页可为警告阈值或严重阈值指定其它更正操作。

单击“Collection Schedule（收集调度）”链接并更改已调度的收集时间间隔。请注意，每个调度都会影响一组度量。

## 创建和测试预警

1. 指定阈值。
2. 创建测试用例。
3. 检查预警。

The screenshot shows three main panels:

- Panel 1 (Top): Edit Advanced Settings: Tablespace Space Used (%)**

Select	Tablespace Name	Comparison Operator	Warning Threshold	Critical Threshold	Corrective Action
<input checked="" type="radio"/> All others	>=	70	75	None	
- Panel 2 (Middle): Show SQL**

```
CREATE TABLE "HR"."FILLER" TABLESPACE "INVENTORY"
STORAGE ( INITIAL 8M) AS SELECT * FROM HR.EMPLOYEES
```
- Panel 3 (Bottom): Alerts**

Severity	Category	Name	Impact	Message	Alert Triggered
×	Tablespaces Full	Tablespace Space Used (%)		Tablespace INVENTORY is 80 percent full	Jun 7, 2007 3:24:05 PM

**ORACLE**

### 创建和测试预警

您还可为特定对象设置阈值。

#### 示例

您决定当 INVENTORY 表空间中使用的空间超过 75% 时需要收到严重预警。（此表空间不允许其数据文件自动进行扩展）。要创建和测试该预警，请执行以下步骤：

- 在 Enterprise Manager 中，导航到“Metrics and Policy Settings（度量和策略设置）”页，然后单击“Tablespace Used (%)（已用表空间百分比 (%)）”阈值的“Edit（编辑）”图标。为表空间设置所需的阈值。
- 在“Tables（表）”页上的“Schema（方案）”选项卡下，创建一个表来测试预警。使用“Define using SQL（使用 SQL 定义）”操作复制现有表。STORAGE 子句中的初始设置为 8 MB，这会使表立即在 10 MB 的 INVENTORY 表空间中分配出 80% 的空间。
- 在收到指明此表无法扩展的错误后，请在“Database（数据库）”主页中查看关联的预警。默认情况下，每 10 分钟会收集一次已用表空间百分比 (%)。

大多数预警都包含相关联指导的名称，调用该指导可获得详细建议。对于每一条相应的预警消息，Database Control 都会提供一个链接以供调用相应的指导。

## 预警通知

Select Metric ▾	Objects	Severity States	Corrective Action States	
			On Critical	On Warning
<input type="checkbox"/> Session Limit Usage (%)	n/a	Critical		
<input type="checkbox"/> Session Terminated Status	n/a	Critical		
<input type="checkbox"/> Tablespace Space Used (%)	All Objects (Tablespace Name)	Critical		
<input type="checkbox"/> Wait Time (%)	n/a	Critical		

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 预警通知

通知机制使用 Enterprise Manager 用户界面。通知机制建立在通知规则概念的基础上，通知规则又确定了一组即将发出预警的相应通知机制。

使用 Database Control 编辑通知规则。在主页上，单击“Preferences（首选项）”链接显示“General（一般信息）”页，在该页中可指定接收通知的电子邮件地址。

在“General（一般信息）”页中，单击“Notification（通知）”区域中的“Rules（规则）”链接。选择“Database Availability and Critical States（数据库可用性和严重状态）”规则，然后单击“Edit（编辑）”按钮。此时会进入“Edit Notification Rule: Database Availability and Critical States（编辑通知规则：数据库可用性和严重状态）”页，在该页单击“Metrics（度量）”选项卡并编辑要接收通知的度量。

## 预警通知（续）

可以指定在出现特定预警时由 Enterprise Manager 直接向您发出通知。例如，如果您指定要接收严重预警的电子邮件通知，并且为每个调用度量的系统响应时间设置了严重阈值，那么，您可以发送一封电子邮件消息，其中包含类似于以下内容的消息：

```
Host Name=mydb.us.mycompany.com
Metric=Response Time per Call
Timestamp=08-NOV-2005 10:10:01 (GMT -7:00)
Severity=Critical
Message=Response time per call has exceeded the threshold.
See the latest ADDM analysis.
Rule Name= Rule
Owner=SYSMAN
```

该电子邮件包含指向主机名及最新 ADDM 分析的链接。

默认情况下，处于严重状态（如“数据库已关闭”、“一般预警日志错误状态”和“已用表空间”）的预警已设置发送通知。但是，要收到这些通知，必须按照以下步骤设置电子邮件信息：

1. 在任何 Database Control 页中，单击页眉和页脚区中的“Setup（设置）”链接。
2. 在“Setup（设置）”页中选择“Notification Methods（通知方法）”。
3. 在“Notifications Methods（通知方法）”页的“Mail Server（邮件服务器）”区域中输入所需的信息。

还有其它通知方法，这包括脚本和简化网络管理协议 (SNMP) 陷阱。后者可用于与第三方应用程序进行通信。

要收到通知，请执行以下操作：

1. 在任何 Database Control 页中，单击页眉和页脚区中的“Preferences（首选项）”链接。
2. 在“Preferences（首选项）”页中选择“General（一般信息）”。在“E-mail Addresses（电子邮件地址）”区域中输入电子邮件地址。
3. 可以有选择地编辑通知规则（如更改接收通知的严重状态）。要执行此操作，请单击“Notification Rules（通知规则）”。此时会出现“Notification Rules（通知规则）”页。

**注：**有关配置通知规则的详细信息，请参阅《Oracle Enterprise Manager 高级配置》文档。

## 对预警作出响应

- 如果需要，应搜集更多的输入信息（如运行 ADDM 或其它指导进行搜集）。
- 调查严重错误。
- 采取纠正措施。
- 确认不会自动清空的预警。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

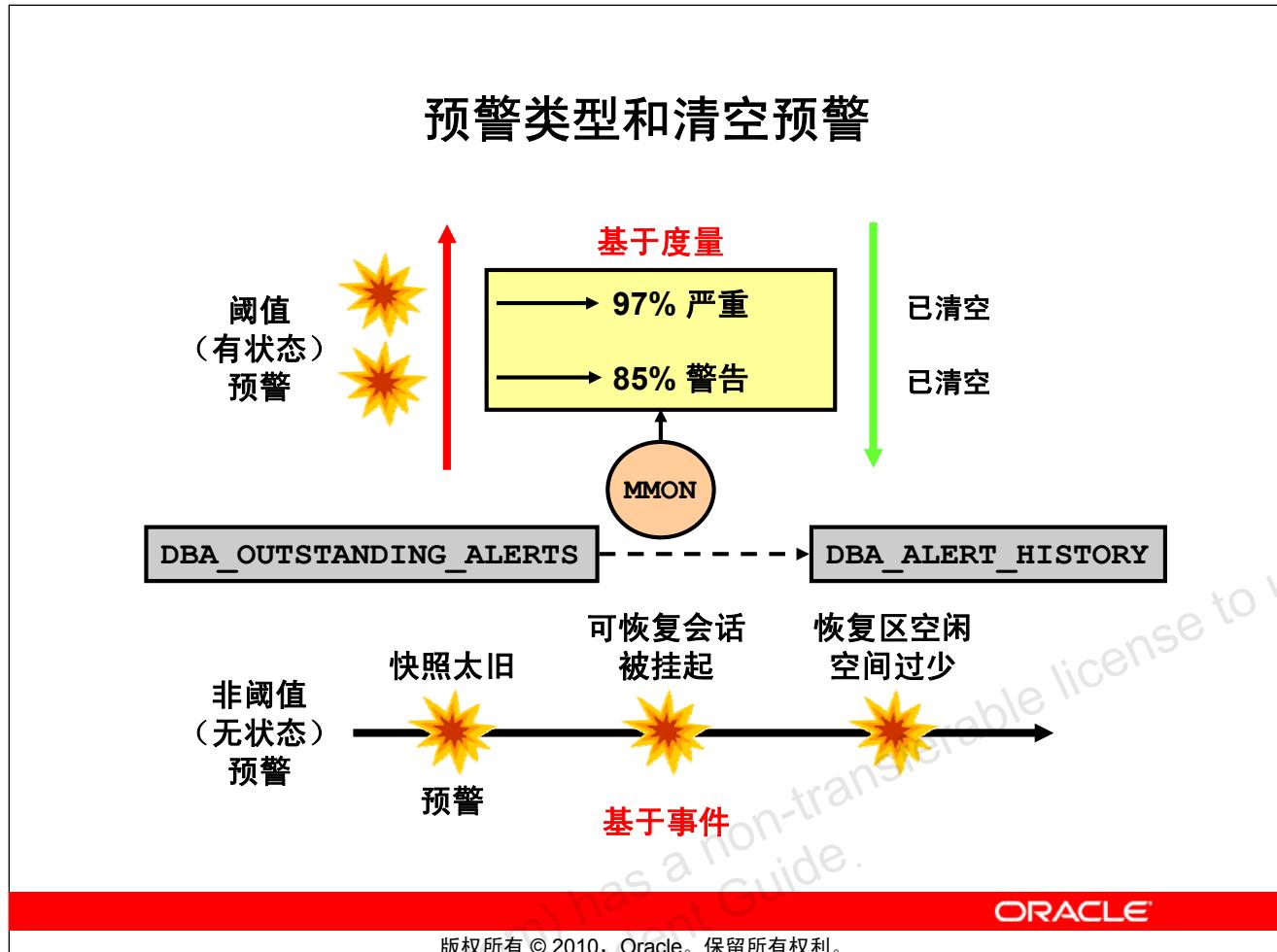
### 对预警作出响应

收到预警时，请按照预警提供的建议进行操作，或者可以考虑运行 ADDM（或其它适当的指导），以便获取有关系统或对象行为的更多详细诊断信息。

系统会为严重错误生成预警和意外事件。严重错误通常会生成收集到问题中的意外事件。可使用“Support Workbench（支持工作台）”进行调查并尽量向 Oracle 技术支持报告问题。

问题的起因消失后，会自动清空大多数预警（如“空间不足”）。但是，如果就其它预警（如“一般预警日志错误”）向您发送通知，则必须由您进行确认。采取必要的纠正措施后，可通过清空或清除预警进行确认。清空某个预警后，会将这个预警发送到预警历史记录，可在“Related Links（相关链接）”下的主页上浏览历史记录。清除一个预警会从预警历史记录中删除它。

要清空类似“一般预警日志错误”的预警，请在“Diagnostic Summary（诊断概要）”下的主页上单击“Alert Log（预警日志）”链接。此时会出现“Alert Log Errors（预警日志错误）”页。选择要清空的预警，然后单击“Clear（清空）”。要清除一个预警，请选择这个预警，然后单击“Purge（清除）”。也可单击“Clear Every Open Alert（清理每个打开的预警）”按钮或“Purge Every Alert（清除每个预警）”按钮。



## 预警类型和清空预警

服务器生成预警分为两种类型：阈值预警和非阈值预警。

大多数服务器生成的预警是通过对数据库度量设置警告阈值和严重阈值来配置的。可为 120 多个度量定义阈值，这些度量包括：

- 每秒的物理读取次数
- 每秒的用户提交次数
- SQL 服务响应时间

除了表空间的空间占用率度量与数据库相关外，其它度量都与实例相关。阈值预警又称为“有状态预警”，清空预警条件后，会自动清空这些预警。有状态预警显示在 DBA\_OUTSTANDING\_ALERTS 中，清空后会转到 DBA\_ALERT\_HISTORY。

服务器生成的其它预警对应于特定的数据库事件，如 ORA-\* 错误、“快照太旧”错误、“恢复区空间过少”和“可恢复会话被挂起”等。这些都不是基于阈值的预警，又称为“无状态预警”。无状态预警会直接转到历史记录表。只在 Database Control 环境中，清空无状态预警才有意义，因为 Database Control 会在自身的资料档案库中存储无状态预警。

## 小测验

诸如 SNAPSHOT TOO OLD 等无状态预警可以在字典视图 DBA\_OUTSTANDING\_ALERTS 中找到。

1. 正确
2. 错误

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小结

在本课中，您应该已经学会：

- 管理优化程序统计信息
- 管理自动工作量资料档案库 (AWR)
- 使用自动数据库诊断监视器 (ADDM)
- 说明和使用指导框架
- 设置预警阈值
- 使用服务器生成的预警
- 使用自动任务

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 12 概览：预先维护

本练习介绍了使用 ADDM 预先管理数据库，其中包括：

- 准备一个问题进行分析
- 复查数据库的性能
- 实施解决方案

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

# 13

性能管理

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 课程目标

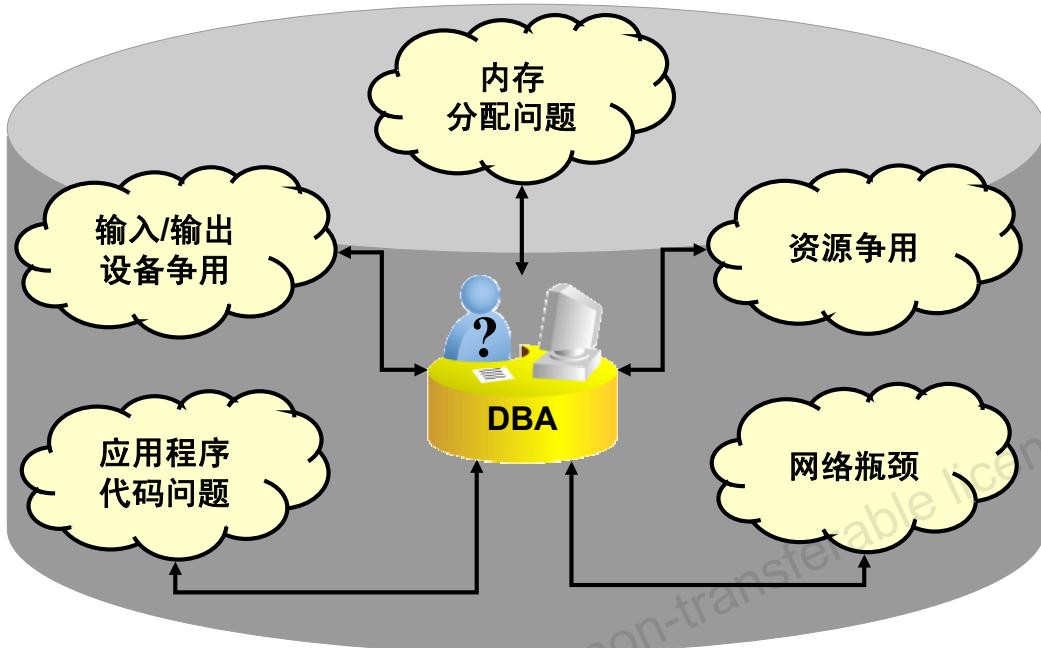
学完本课后，应能完成以下工作：

- 使用 Oracle Enterprise Manager 监视性能
- 使用自动内存管理 (AMM)
- 使用内存指导调整内存缓冲区的大小
- 查看与性能相关的动态视图
- 排除无效和不可用对象产生的故障

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 性能监视



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

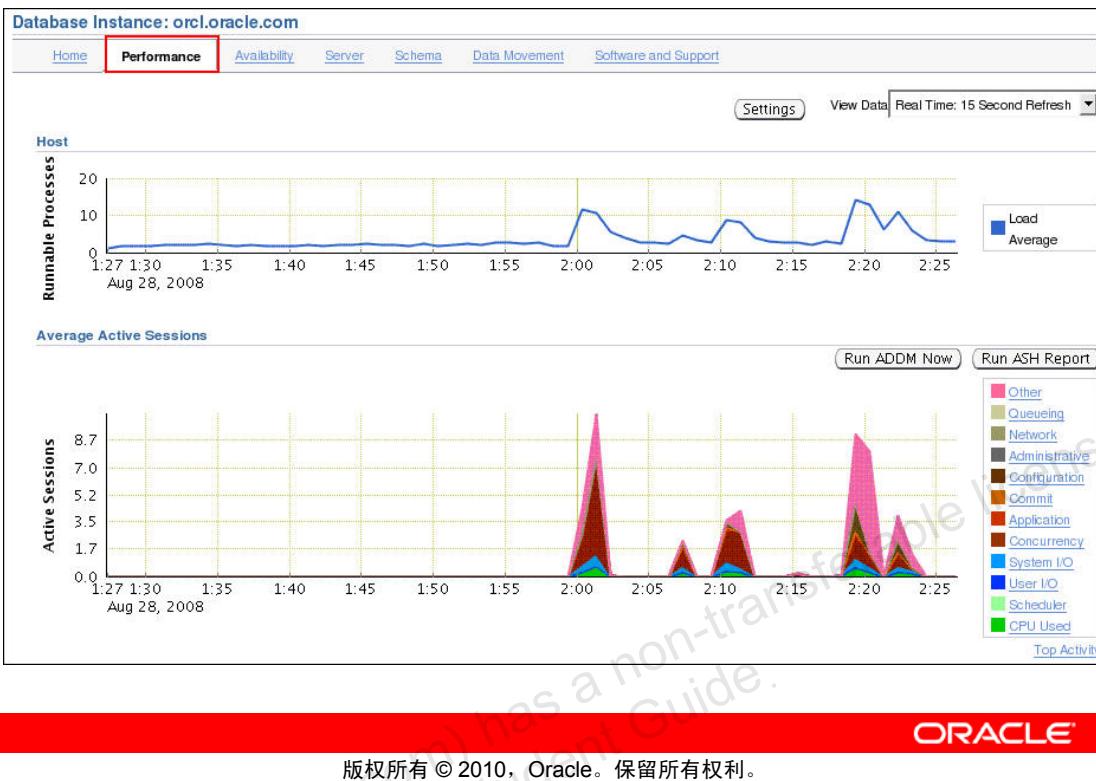
### 性能监视

要管理 Oracle Database 11g 并保持其正常运行，数据库管理员 (DBA) 必须通过定期监视性能来查找瓶颈所在，然后更正问题。

DBA 可以查看的性能度量有数百种，包括从网络性能和磁盘输入/输出 (I/O) 速度到运行单个应用程序操作所花费的时间等所有相关方面。这些性能度量通常称为“数据库度量”。

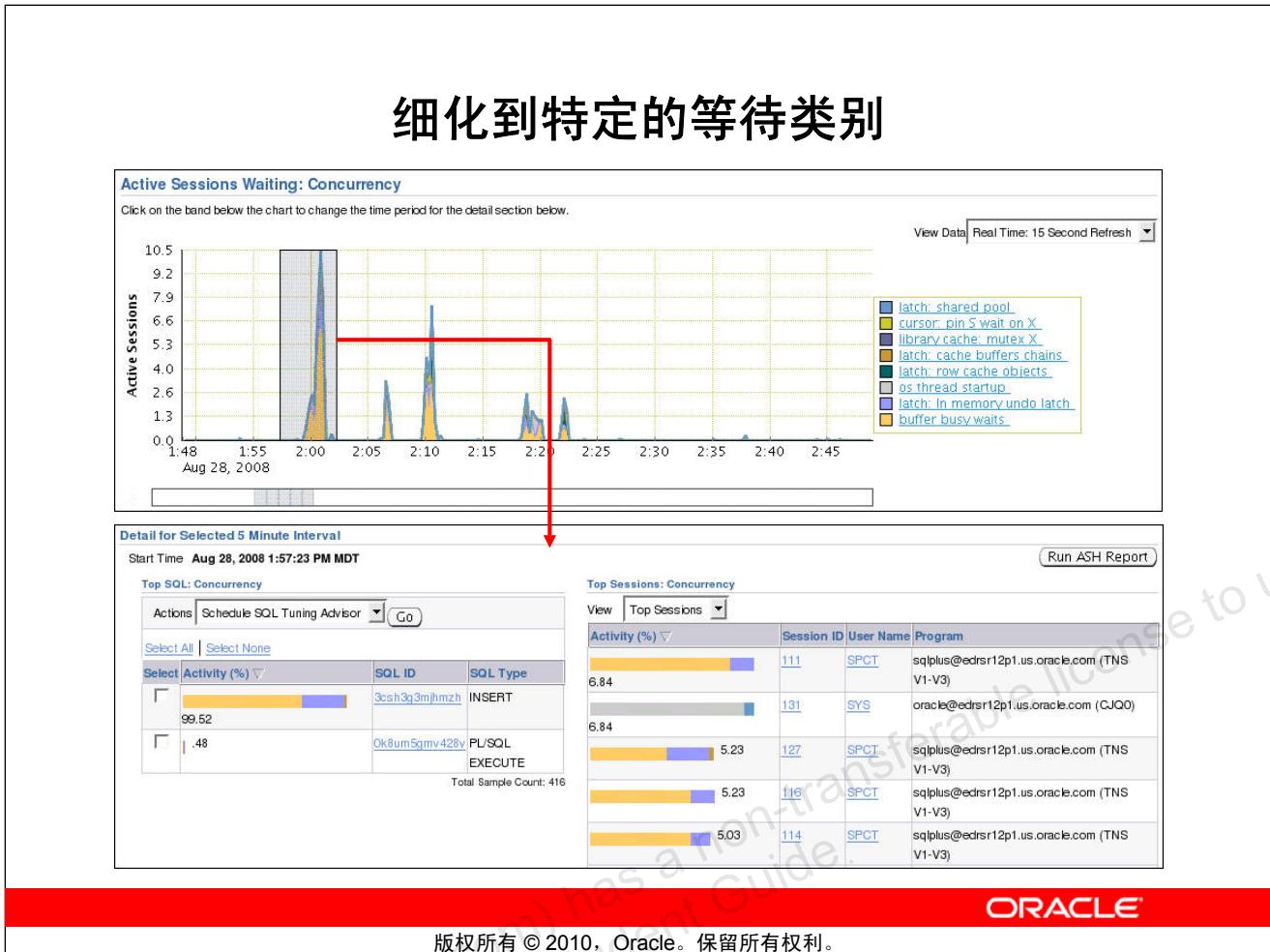
注：有关 Oracle DB 性能的详细信息，请参阅“Oracle Database 11g：性能优化”课程。

# Enterprise Manager 的性能页



## Enterprise Manager 的性能页

Enterprise Manager 中的“Performance（性能）”页是一个门户，它提供了一组功能强大的性能监视和优化工具。此页的第一组图概述了进程和活动会话活动。“Average Active Sessions（平均活动会话数）”图中显示了 CPU 使用级别及造成大多数等待事件的资源。在幻灯片中，可以看到“Concurrency（并发）”和其它等待时间最近都有所增加。在这些峰值的每个位置，系统 I/O 和 CPU 使用率也稍有增加。单击这些类别可查看有关等待时间的详细信息。I/O 数据被分为输入类型和输出类型（例如日志文件读取、控制文件写入等）。



### 细化到特定的等待类别

细化到特定的等待类别后，可查看具体到五分钟时间间隔的详细资料，同时可以看见这段时间内与特定的等待事件相关联的顶级运行 SQL 和顶级运行会话。这样可对系统执行速度下降的问题进行事后分析，从而确定可能存在的原因。

幻灯片中的示例显示了对上一页幻灯片的活动会话图形的“Concurrency（并发）”类别进行细化的结果。

## 性能页：吞吐量

向下滚动“Performance（性能）”页面。

The screenshot shows the Oracle Database Performance page for the instance `orcl.oracle.com`. The top navigation bar includes Home, Performance (which is selected), Availability, Server, Schema, Data Movement, and Software and Support. Below the navigation is a sub-navigation bar with Throughput, I/O, Parallel Execution, and Services. The main content area displays two line charts for Aug 28, 2008, from 1:55 to 2:54. The first chart, 'Instance Throughput Rate' (Per Second), shows Logons (blue line) and Transactions (green line). The second chart, 'Physical Reads (KB)' and 'Redo Size (KB)', shows Physical Reads (blue line) and Redo Size (green line). Both charts show a significant peak around 2:19.

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 性能页：吞吐量

单击“Performance（性能）”主页上的“Throughput（吞吐量）”和“I/O”选项卡，可以查看实例吞吐量图和实例磁盘 I/O 图。幻灯片中选择的是“Throughput（吞吐量）”选项卡。

## 性能监视：顶级会话

**Top Consumers**

Collected From Jul 31, 2009 8:40:17 AM GMT+07:00 To Jul 31, 2009 8:40:32 AM GMT+07:00

Overview Top Services Top Modules Top Actions Top Clients **Top Sessions** Show Active SQL Customize

(Kill Session) View Disable SQL Trace Enable SQL Trace

Select	SID	DB User	CPU (1/100 sec)	PGA (bytes)	Physical Reads	Logical Reads	Hard Parses	Total Parses	Disk Sorts	Status	Program	Module	OS PID	Machine	OS User	SQL Trace
<input checked="" type="radio"/>	36	INVENTORY	1 1540712	20	228	0	0	0	0	ACTIVE	sqlplus@edrsr12p1.us.oracle.com	SQL*Plus	18269	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	59	DBSNMP	12 7168500	0	56	0	10	0	0	ACTIVE	emagent@edrsr12p1.us.oracle.com	emagent_SQL_database	9753	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	42	CJQ0	0 1794548	0	6	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com	(CJQ0)	13001	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	41	DBSNMP	0 2384372	0	3	0	2	0	0	ACTIVE	emagent@edrsr12p1.us.oracle.com	emagent_AgMetrics	13286	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	29	SYSMAN	0 2843124	0	1	0	5	0	0	ACTIVE	OMS	OEM.DefaultPool	12965	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	49	DBSNMP	14 1532404	0	0	0	2	0	0	ACTIVE	OMS	Realtime Connection	18390	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	34	SYSMAN	0 2384372	0	0	0	0	0	0	ACTIVE	OMS	OEM.SystemPool	13061	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	30	SYSMAN	0 3105268	0	0	0	3	0	0	ACTIVE	OMS	OEM.DefaultPool	12967	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	18	MMNL	0 1270260	0	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com	(MMNL)	12861	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	25	QMNC	0 745972	0	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com	STREAMS	12963	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	57	J001	0 418292	0	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com	(J001)	18451	edrsr12p1.us.oracle.com	oracle	DISABLED
<input type="radio"/>	43	SMCO	0 418292	0	0	0	0	0	0	ACTIVE	oracle@edrsr12p1.us.oracle.com	KTSJ	13544	edrsr12p1.us.oracle.com	oracle	DISABLED

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 性能监视：顶级会话

单击“Additional Monitoring Links（其它监视链接）”部分中的“Top Consumers（顶级使用者）”会进入“Top Consumers（顶级使用者）”页。

“Top Consumers Overview（顶级使用者概览）”页以图形格式显示以下内容：

- 顶级服务
- 顶级模块（按服务统计）
- 顶级操作（按服务和模块统计）
- 顶级客户机

在“Top Consumers（顶级使用者）”页上单击“Top Sessions（顶级会话）”选项卡，可查看占用资源最多的会话的重要统计信息：

- CPU
- PGA 内存
- 逻辑读取数
- 物理读取数
- 硬性语法分析计数
- 排序计数

## 性能监视：顶级会话（续）

单击某个列名，将按照该列的值对结果进行排序。

此页上的表中列出了按逻辑读取数排序的会话。其中显示了会话 36 中的用户 INVENTORY 正在产生这段特定时间内最大的逻辑读取数。

## 性能监视：顶级服务

The screenshot shows the Oracle Database 11g Performance Monitor interface. At the top, there are tabs: Overview, Top Services (which is selected), Top Modules, Top Actions, Top Clients, and Top Sessions. Below the tabs, there's a dropdown menu set to 'Active Services' and buttons for 'Enable SQL Trace', 'Disable SQL Trace', and 'View SQL Trace File'. There are also 'Select All' and 'Select None' checkboxes.

Select Service	Activity (% for the last 5 minutes)	SQL Trace Enabled	Delta Elapsed Time (seconds)	Cumulative Elapsed Time (seconds)
<input type="checkbox"/> SYS\$USERS	42.9	FALSE	0	227
<input type="checkbox"/> SYS\$BACKGROUND	35.7	FALSE	0	0
<input type="checkbox"/> SH	14.3	FALSE	0	2
<input type="checkbox"/> SERV1	7.1	FALSE	0	2

Delta CPU Time (seconds)	Cumulative CPU Time (seconds)	Physical I/O (blocks)	Cumulative Physical I/O (blocks)
0	0	0	16031
0	137	0	14414
0	1	15	637
0	2	0	12

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 性能监视：顶级服务

在多层系统中，如果存在作为共享数据库连接的应用程序服务器，就不能通过查看会话来获得分析性能所需要的信息。将会话按服务名称分组可以更加准确地监视性能。

在幻灯片示例中，有四个服务：SYS\$USERS、SYS\$BACKGROUND、SH 和 SERV1。无论特定请求使用什么会话，只要通过其中一个服务连接会话，会话的性能数据就会被捕获并列在该服务名之下。对于显示的应用程序服务（SH 和 SERV1），可以从列表中清楚地看到 SH 服务在此五分钟的时间间隔内较为活跃。

## 管理内存组件

- 自动内存管理 (AMM)
  - 使您可以指定分配给实例的总内存（包括 SGA 和 PGA）
- 自动共享内存管理 (ASMM):
  - 使您可以通过一个初始化参数指定 SGA 总内存
  - 使 Oracle 服务器可以管理分配给共享池、Java 池、缓冲区高速缓存、流池和大型池的内存量
- 手动设置共享内存管理:
  - 通过多个单独的初始化参数调整组件的大小
  - 使用适当的内存指导提出建议



版权所有 © 2010, Oracle。保留所有权利。

### 管理内存组件

在 Oracle Database 11g 中，可以指定分配给实例的总内存。可以根据需要，在系统全局区 (SGA) 和程序全局区 (PGA) 之间动态地重新分配内存。此方法称为自动内存管理 (AMM)，它仅适用于支持动态释放内存的平台。这样就简化了内存管理任务。

内存指导可帮助您在各种级别设置初始化参数。具体使用哪个指导将取决于要在哪个级别指定内存参数。如果启用了 AMM，则只能使用内存大小指导。

使用自动共享内存管理 (ASMM) 可以将 SGA 作为整体进行管理。SGA 包含多个组件。其中许多组件的大小都可在初始化参数限制范围内动态地进行调整，以便获得最佳性能。启用 AMM 后，ASMM 也会自动启用。如果只启用 ASMM 而未启用 AMM，则可使用 SGA 大小指导。

通过设置各组件的初始化参数，可以手动管理该组件的大小。如果 Oracle 服务器通知您出现了与 SGA 或 PGA 组件大小相关的性能问题，可以使用该组件的内存指导来确定适当的新设置。内存指导可以模拟参数更改的效果。

**启用自动内存管理 (AMM)**

Database Instance: orcl.oracle.com > Advisor Central >

Logged In As S...

Memory Advisors

单击“Enable (启用)”可启用自动内存管理。

Automatic Memory Management **Disabled** **Enable**

Memory Advisors

When Automatic Memory Management is enabled, the database will automatically set the optimal distribution of memory will change from time to time to accomodate changes in the workload.

Automatic Memory Management **Enabled** **Disable**

Total Memory Size 556 MB **Advice**

Maximum Memory Size 1000 MB

The database must be restarted before any changes to this value take effect.

使用内存大小指导。

Memory Size Advice

Improvement in DB Time (%)

Total Memory Size (MB)

Percentage improvement in DB Time for various sizes of Total Mem  
X Total Memory Size  
■ Maximum Memory Size

Total Memory Size (MB) 556

You can click on the curve in the graph to set a new value. Total Memory Size cannot be greater than the Maximum Memory Size. First modify the Maximum Memory size (from the parent page) and then select a value of Total Memory up to the Maximum Memory size.

Cancel OK

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 启用自动内存管理 (AMM)

如果在配置数据库时未启用自动内存管理 (AMM)，可以执行以下步骤来启用它：

1. 在“Database (数据库)”主页上，单击“Server (服务器)”选项卡。
  2. 在“Database Configuration (数据库配置)”区域，单击“Memory Advisors (内存指导)”。
- 此时将显示“Memory Advisors (内存指导)”页。
3. 单击“Enable (启用)”启用自动内存管理。
- 此时会出现“Enable Automatic Memory Management (启用自动内存管理)”页。
4. 为自动内存管理的“Total Memory Size (内存总大小)”和“Maximum Memory Size (最大内存大小)”设置适当的值。  
**注：**如果更改“Maximum Memory Size (最大内存大小)”，必须重新启动数据库实例。
5. 单击“OK (确定)”。

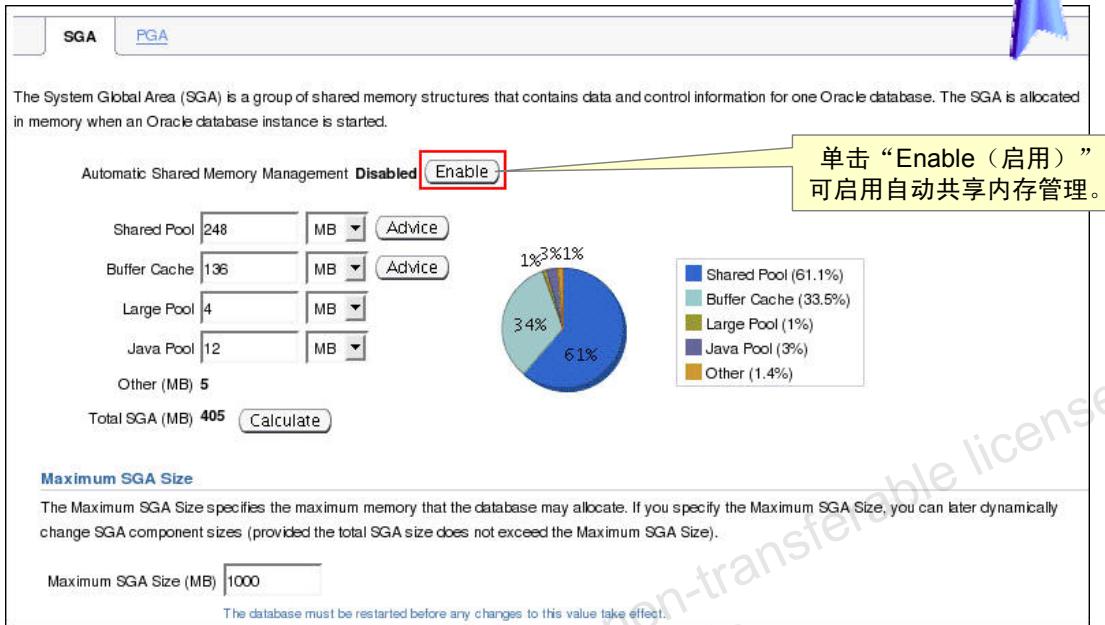
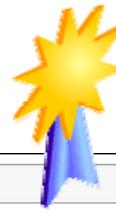
以后可通过增大“Total Memory Size (内存总大小)”字段或初始化参数 MEMORY\_TARGET 的值来增大内存总大小。但不能将该大小设置为大于“Maximum Memory Size (最大内存大小)”字段或 MEMORY\_MAX\_TARGET 参数所指定的值。有关详细信息，请参阅《Oracle 数据库管理员指南》。

## 启用自动内存管理 (AMM) (续)

启用 AMM 后，可使用内存大小指导来帮助调整最大内存大小和目标内存大小。

注：Oracle 建议使用自动内存管理来简化内存管理任务。

# 启用自动共享内存管理 (ASMM)



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 启用自动共享内存管理 (ASMM)

如果已启用 AMM，则会自动启用自动共享内存管理。如果在配置数据库时未启用 AMM 或 ASMM，可以执行以下步骤来启用自动共享内存管理：

1. 在“Database（数据库）”主页上，单击“Server（服务器）”选项卡。
2. 在“Database Configuration（数据库配置）”区域，单击“Memory Advisors（内存指导）”。
3. 此时将显示“Memory Advisors（内存指导）”页。
4. 向下滚动到 SGA 部分。单击“Enable（启用）”启用自动共享内存管理。

此时会出现“Enable Automatic Shared Memory Management（启用自动共享内存管理）”页。

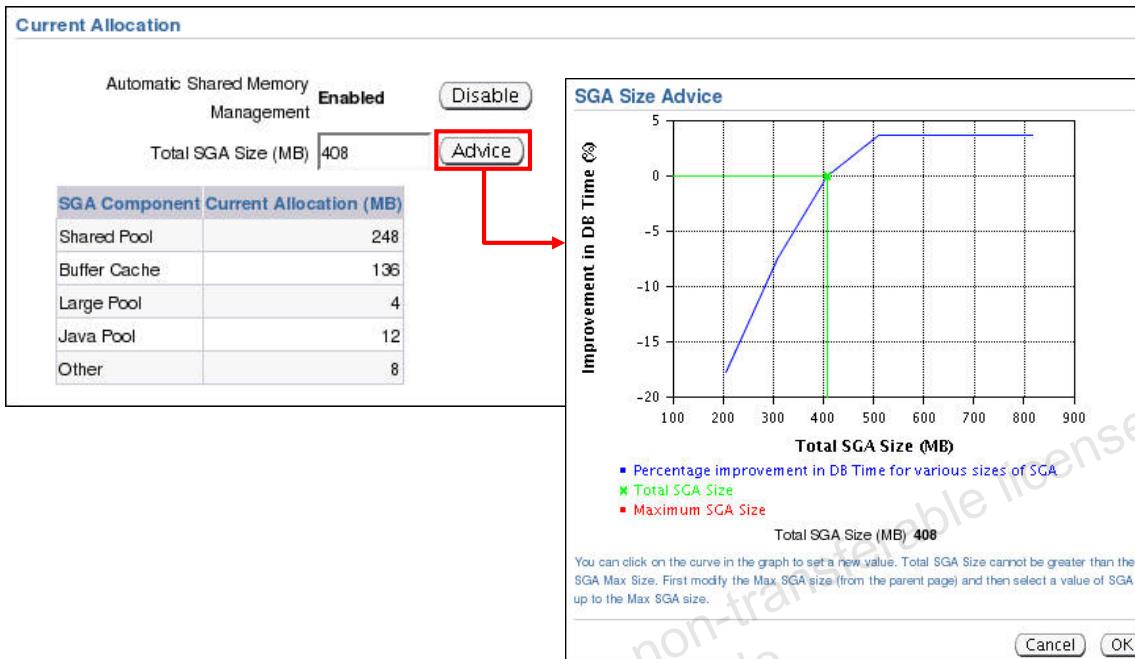
4. 指定 SGA 总大小。单击“OK（确定）”。

以后可通过增大“Total SGA Size（SGA 总大小）”字段或初始化参数 SGA\_TARGET 的值来增大 SGA 的总大小。但不能将该大小设置为大于“Maximum SGA Size（最大 SGA 大小）”字段或 SGA\_MAX\_SIZE 参数所指定的值。有关详细信息，请参阅《Oracle 数据库管理员指南》。

禁用 AMM 时，也可以访问 PGA 指导。建议使用 PGA 指导设置 PGA 内存值。单击“PGA”选项卡可访问 PGA 属性页。单击“Advice（建议）”调用 PGA 指导。

注：Oracle 建议使用自动共享内存管理来简化内存管理任务。

## 自动共享内存指导



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 自动共享内存指导

如果启用了 ASMM，则不应为 ASMM 所管理的特定共享内存组件设置初始化参数。启用 AMM 后，可使用 SGA 大小指导帮助您选择 SGA 总大小的最佳值。

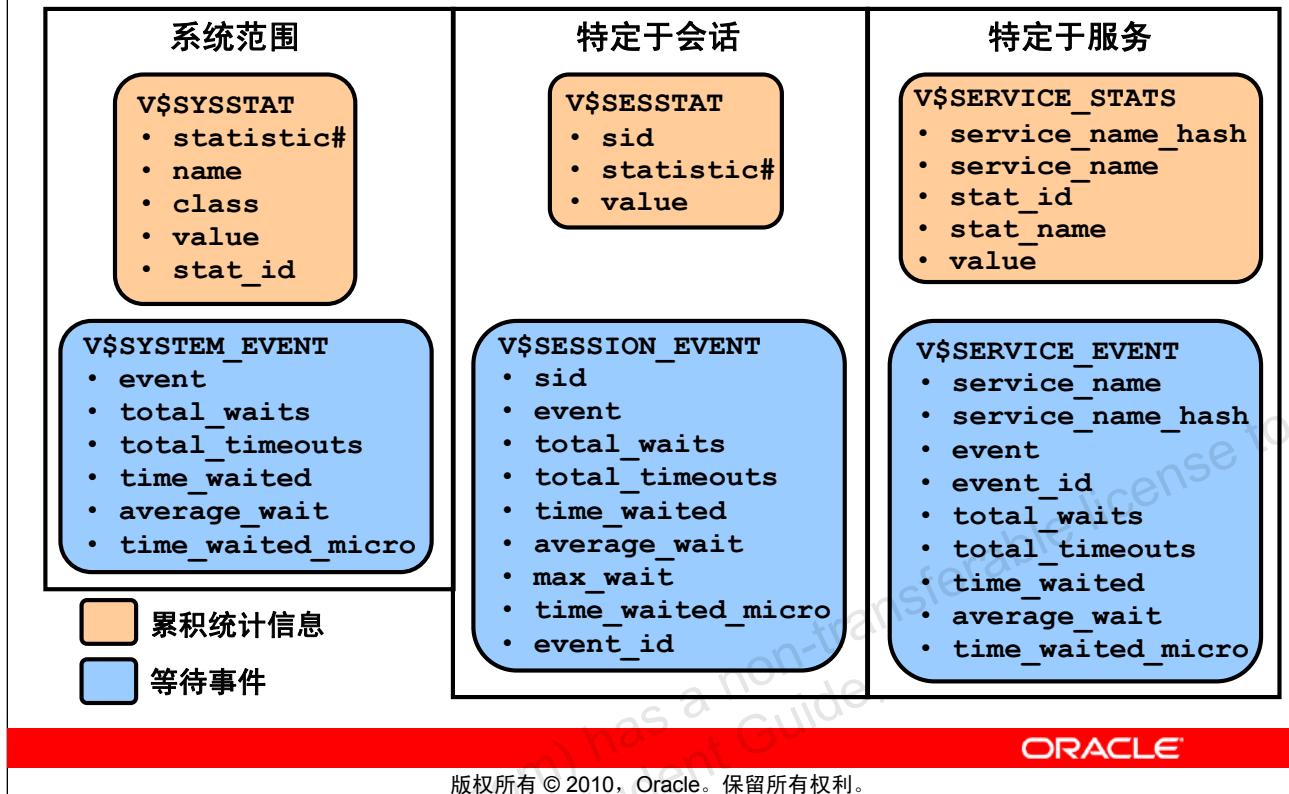
启用 ASMM 之前，应该从 SPFILE 中删除单个内存区参数，因为设置这些参数会使 ASMM 受到限制。看到 ASMM 分配的效果之后，如果决定要调整特定组件的分配值，可以为这些组件指定值。如果指定的值比目前的值小，这些值将被视为各个组件的最小内存大小。如果指定的值比目前的值大，那么只要有空闲内存，就可以将内存组件的大小重新上调至您提供的值。执行此操作会限制可用于自动调整的内存量，但在环境需要对大小进行特殊调整、而 ASMM 又无法满足此需要时，可使用此功能。

相关的初始化参数如下：

- SHARED\_POOL\_SIZE
- LARGE\_POOL\_SIZE
- JAVA\_POOL\_SIZE
- DB\_CACHE\_SIZE
- STREAMS\_POOL\_SIZE

要在启用 ASMM 后调整这些参数，必须使用 ALTER SYSTEM 命令。

## 动态性能统计信息



### 动态性能统计信息

要有效地诊断性能问题，必须使用统计信息。Oracle 服务器会在不同粒度级生成多种类型的统计信息。

在系统范围级、会话级和服务级，均会计算出等待事件和累积统计信息。在幻灯片中，视图的顶行显示累积统计信息，底行则显示等待事件的视图。

分析其中任一范围内的性能问题时，通常会查看感兴趣的某一时段内统计信息的变化（差值）。所有可能的等待事件都会列入 V\$EVENT\_NAME 视图目录。所有统计信息均编录在 V\$STATNAME 视图中；Oracle DB 中大约有 480 种统计信息。

## 动态性能统计信息（续）

### 显示系统范围的统计信息

示例：

```
SQL> SELECT name, class, value FROM v$sysstat;
NAME                      CLASS      VALUE
-----
...
table scans (short tables)    64        135116
table scans (long tables)    64         250
table scans (rowid ranges)   64          0
table scans (cache partitions) 64         3
table scans (direct read)    64          0
table scan rows gotten       64     14789836
table scan blocks gotten     64      558542
...
...
```

系统范围的统计信息按优化主题和调试目的分类。这些类别包括一般实例活动、重做日志缓冲区活动、锁定、数据库缓冲区高速缓存活动等等。

## 故障排除和优化视图

### 实例/数据库

V\$DATABASE  
V\$INSTANCE  
V\$PARAMETER  
V\$SPPARAMETER  
V\$SYSTEM\_PARAMETER  
V\$PROCESS  
V\$BGPROMISE  
V\$PX\_PROCESS\_SYSSTAT  
V\$SYSTEM\_EVENT

### 磁盘

V\$DATAFILE  
V\$FILESTAT  
V\$LOG  
V\$LOG\_HISTORY  
V\$DBFILE  
V\$TEMPFILE  
V\$TEMPSEG\_USAGE  
V\$SEGMENT\_STATISTICS

### 内存

V\$BUFFER\_POOL\_STATISTICS  
V\$LIBRARYCACHE  
V\$SGAINFO  
V\$PGASTAT

### 争用

V\$LOCK  
V\$UNDOSTAT  
V\$WAITSTAT  
V\$LATCH

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 故障排除和优化视图

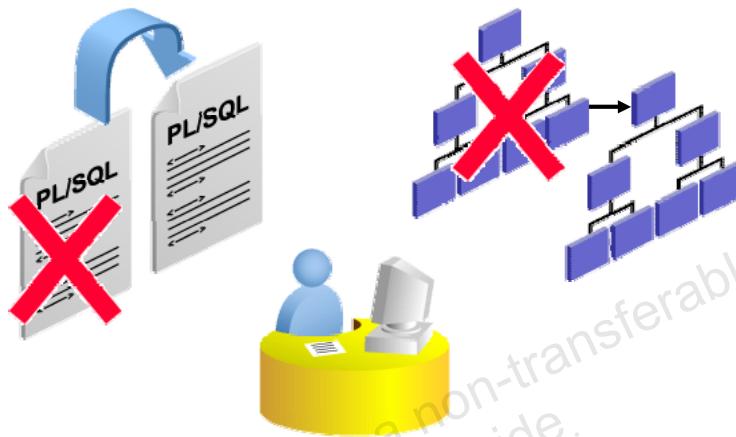
幻灯片中列出了部分视图，这些视图有助于确定导致性能问题的原因或分析数据库的当前状态。

有关这些视图的完整说明，请参阅《Oracle 数据库参考》。

## 无效和不可用对象

对性能的影响：

- PL/SQL 代码对象要重新编译。
- 索引要重建。



ORACLE

版权所有 © 2010, Oracle。保留所有权利。

### 无效和不可用对象

无效 PL/SQL 对象和不可用索引会对性能产生影响。无效 PL/SQL 对象必须先重新编译，然后才能使用。这需要将编译时间加到尝试访问 PL/SQL 程序包、过程或函数的第一个操作中。如果 PL/SQL 重编译未成功，则操作会因发生错误而失败。优化程序会忽略不可用索引。如果 SQL 语句性能的好坏取决于已标记为不可用的索引，则只有重建索引才能改善性能。

**无效 PL/SQL 对象：**可通过查询数据字典来查看 PL/SQL 对象的当前状态。可使用以下命令行查找到无效 PL/SQL 对象：

```
SELECT object_name, object_type FROM DBA_OBJECTS
WHERE status = 'INVALID';
```

默认情况下，每 24 小时会检查一次“所有者的无效对象计数”度量。如果单个所有者的对象数超过两个，就会发出预警。

如果查找到 INVALID 状态的 PL/SQL 对象，需要回答的第一个问题是“此对象曾经是 VALID 的吗？”应用程序开发人员常常会忘记清除不起作用的代码。如果 PL/SQL 对象因代码错误而无效，则除了纠正代码错误之外，没有什么别的方法。如果该过程在过去曾经有效，最近才变为无效，则可选择以下两种方法解决这个问题：

## 无效和不可用对象（续）

- 不做任何处理。如果需要，大多数 PL/SQL 对象在被调用时会自动重新编译。用户在对象重新编译时会经历短暂的延迟。（大多数情况下，这种延迟不十分明显。）
- 手动重新编译无效对象。

使用 Enterprise Manager 或通过 SQL 命令可手动重新编译无效的 PL/SQL 对象：

```
ALTER PROCEDURE HR.add_job_history COMPILE;
```

手动重新编译 PL/SQL 数据包，需要执行以下两个步骤：

```
ALTER PACKAGE HR.maintainemp COMPILE;
ALTER PACKAGE HR.maintainemp COMPILE BODY;
```

**不可用索引：**可通过查询 DBA\_INDEXES 数据字典视图查找无效索引：

```
SELECT index_name, table_name FROM DBA_INDEXES
WHERE status = 'UNUSABLE';
```

对于分区索引，状态保存在 DBA\_IND\_PARTITIONS 视图中。

通过重建不可用索引来重算指针，可使不可用索引变为有效。重建不可用索引会在新位置重新创建索引，然后会删除不可用索引。使用 Enterprise Manager 或通过 SQL 命令可完成此操作：

```
ALTER INDEX HR.emp_empid_pk REBUILD;
ALTER INDEX HR.emp_empid_pk REBUILD ONLINE;
ALTER INDEX HR.email REBUILD TABLESPACE USERS;
```

如果省略了 TABLESPACE 子句，则会在索引曾经所在的相同表空间中重建索引。使用 REBUILD ONLINE 子句，用户可以在重建时继续更新索引表。（如果不使用 ONLINE 关键字，用户必须等待重建完成后才能对受影响的表执行 DML。如果索引不可用，即便使用了 ONLINE 关键字，也不会在重建过程中使用该索引。）

Enterprise Manager 使用“Reorganize（重组）”操作修复 UNUSABLE 索引。

**注：**重建索引时需要有可用于重建的空闲空间。请在尝试重建前验证是否有足够的空间。Enterprise Manager 会自动检查空间要求。

## 小测验

自动内存管理允许 Oracle 实例将内存从 \_\_\_\_\_ 重新分配至 SGA。

1. 大型池
2. 日志缓冲区
3. PGA
4. 流池

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 3

## 小测验

SGA\_TARGET 的大小不能大于 \_\_\_\_\_。

1. LOG\_BUFFER
2. SGA\_MAX\_SIZE
3. STREAMS\_POOL\_SIZE
4. PGA\_AGGREGATE\_TARGET

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

答案: 2

## 小结

在本课中，您应该已经学会：

- 使用 Oracle Enterprise Manager 监视性能
- 使用自动内存管理 (AMM)
- 使用内存指导调整内存缓冲区的大小
- 查看与性能相关的动态视图
- 排除无效和不可用对象产生的故障

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

## 练习 13 概览：监视和改进性能

本练习包含以下主题：

- 检测和修复不可用索引
- 使用 Enterprise Manager 中的“Performance（性能）”页

ORACLE

版权所有 © 2010, Oracle。保留所有权利。

