

Oracle Database 11g: 数据库 管理 — 课堂练习 II

学生指南第 1 册

D50079CN20

版本 2.0

2011 年 3 月

D71767

ORACLE®

作者

Maria Billings

技术撰稿人和审稿人

Christian Bauwens

Yanti Chang

Timothy Chien

Joe Fong

Andy Fortunak

Gerlinde Frenzen

Mark Fuller

Peter Fusek

Joel Goodman

Vimala Jacob

Dominique Jeunot

Pete Jones

Fukue Kawabe

Donna Keesling

Sean Kim

Achiel Langers

Gwen Lazenby

Jerry Lee

Deidre Matishak

Bill Millar

Lakshmi Naraparreddi

Ira Singer

Ranbir Singh

James Spiller

Matt Taylor

Branislav Valny

Jean-Francois Verrier

编辑

Nita Pavitrnan

Raj Kumar

制图员

Satish Bettogowda

出版商

Jayanthi Keshavamurthy

版权所有 © 2011, 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/或其子公司的注册商标。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

建议的日程表 I-3

Oracle Database 11g: “g” 代表 Grid (网格) I-4

用于单实例的 Oracle Grid Infrastructure I-6

课程示例: HR 示例方案 I-8

1 Oracle DB 的核心概念和工具

课程目标 1-2

列出 Oracle DB Server 的核心组件的名称 1-3

Oracle DB Server 体系结构概览 1-4

实例数据库配置 1-6

列出 Oracle DB 的内存结构的名称 1-7

Oracle DB 内存结构 1-8

进程体系结构 1-10

进程结构 1-11

添加进程名称 1-13

进程启动顺序 1-14

数据库存储体系结构 1-15

逻辑和物理数据库结构 1-17

自动存储管理 1-19

ASM 存储组件 1-20

ASM 实例 1-21

DBA 配置工具 1-23

管理框架和相关 DBA 工具 1-24

使用 Oracle Restart 简化数据库管理 1-25

测验 1-27

小结 1-28

2 进行配置以实现可恢复性

课程目标 2-2

备份和恢复功能的用途 2-3

典型的备份和恢复任务	2-4
Oracle 备份和恢复解决方案	2-5
Oracle 备份解决方案	2-6
术语回顾	2-7
您已了解的内容：Oracle 建议的备份	2-9
使用 Recovery Manager	2-10
RMAN 命令的类型	2-11
作业命令：示例	2-12
配置数据库进行备份和恢复操作	2-13
ARCHIVELOG 模式	2-14
配置 ARCHIVELOG 模式	2-15
配置归档日志目标位置	2-16
确保归档日志成功	2-18
指定保留策略	2-20
恢复窗口保留策略：示例	2-22
使用快速恢复区	2-23
定义快速恢复区	2-25
快速恢复区空间管理	2-26
快速恢复区空间使用情况	2-28
自动为您完成的任务	2-30
监视 FRA	2-31
使用快速恢复区的优点	2-32
测验	2-33
小结	2-35
练习 2 概览：进行配置以实现可恢复性	2-36

3 使用 RMAN 恢复目录

课程目标	3-2
RMAN 资料档案库数据存储：比较选项	3-3
在恢复目录中存储信息	3-4
使用恢复目录的原因	3-5
创建恢复目录：三个步骤	3-6
配置恢复目录数据库	3-7
创建恢复目录所有者	3-8
创建恢复目录	3-9
管理恢复目录中的目标数据库记录	3-10

在恢复目录中注册数据库	3-11
使用 Oracle Enterprise Manager 注册数据库	3-12
从恢复目录中注销目标数据库	3-13
将其它备份文件列入目录	3-14
恢复目录重新同步：概念	3-16
手动重新同步恢复目录	3-17
使用 RMAN 存储脚本	3-18
执行 RMAN 存储脚本	3-19
维护 RMAN 存储脚本	3-20
备份恢复目录	3-21
重新创建无法恢复的恢复目录	3-22
导出和导入恢复目录	3-23
升级和删除恢复目录	3-24
IMPORT CATALOG 命令	3-25
创建和使用虚拟专用目录	3-27
使用 RMAN 虚拟专用目录	3-28
恢复目录概要	3-30
测验	3-32
小结	3-34
练习 3 概览：使用 RMAN 恢复目录	3-35
4 配置备份设置	
课程目标	4-2
配置 RMAN 的永久性设置	4-3
查看永久性设置	4-4
控制文件自动备份	4-5
管理永久性设置	4-7
使用介质管理器	4-8
指定备份目标	4-10
配置和分配通道	4-11
创建双向备份集	4-12
使用 CONFIGURE BACKUP COPIES 创建双向备份集	4-13
备份优化	4-14
通过压缩未使用的块节省备份空间	4-16
压缩备份	4-17
使用 RMAN 备份压缩	4-18

加密备份 4-19
 测验 4-20
 小结 4-22
 练习 4 概览：配置备份规范 4-23

5 使用 RMAN 创建备份

课程目标 5-2
 创建备份集 5-3
 创建映像副本 5-4
 创建整体数据库备份 5-6
 RMAN 备份类型 5-8
 快速增量备份 5-10
 启用快速增量备份 5-11
 监视块更改跟踪 5-12
 执行代理副本 5-13
 使用 BACKUP COPIES 创建双向备份集 5-14
 创建备份集的备份 5-15
 备份只读表空间 5-16
 为超大型文件配置备份和还原 5-17
 创建 RMAN 多部分备份 5-18
 归档备份：概念 5-19
 使用 EM 创建归档备份 5-21
 使用 RMAN 创建归档备份 5-22
 管理归档数据库备份 5-23
 备份恢复文件 5-24
 管理备份：报告 5-25
 管理备份：动态性能视图 5-27
 使用 Oracle Enterprise Manager 查看备份报告 5-28
 管理备份：交叉检查和删除 5-29
 测验 5-30
 小结 5-32
 练习 5 概览：创建备份 5-33

6 还原和恢复任务

课程目标 6-2
 还原和恢复 6-3

文件丢失的原因 6-4
关键性与非关键性 6-5
自动恢复临时文件 6-6
日志组状态：综述 6-7
在丢失了重做日志组后进行恢复 6-8
清除日志文件 6-9
丢失了索引表空间后进行恢复 6-10
重新创建索引 6-11
面向数据库管理员的验证方法 6-13
重新创建口令验证文件 6-14
比较完全恢复和不完全恢复 6-16
完全恢复过程 6-17
时间点恢复过程 6-18
恢复只读表空间 6-20
恢复 NOLOGGING 数据库对象 6-21
在丢失了所有控制文件副本后进行恢复：概览 6-22
将控制文件恢复到默认位置 6-23
测验 6-24
小结 6-26

7 使用 RMAN 执行恢复

课程目标 7-2
使用 RMAN RESTORE 和 RECOVER 命令 7-3
执行完全恢复：在 ARCHIVELOG 模式下丢失了非关键数据文件 7-5
执行完全恢复：在 ARCHIVELOG 模式下丢失了系统关键数据文件 7-6
恢复映像副本 7-7
恢复映像副本：示例 7-8
执行到映像副本的快速切换 7-9
使用 SET NEWNAME 切换文件 7-10
SET NEWNAME 的替代变量 7-11
在 NOARCHIVELOG 模式下执行数据库还原和恢复 7-12
使用还原点 7-13
执行时间点恢复 7-14
使用备份控制文件执行恢复 7-16
恢复丢失的服务器参数文件 7-17
从控制文件自动备份还原服务器参数文件 7-18

从自动备份还原控制文件	7-19
使用增量备份恢复处于 NOARCHIVELOG 模式的数据库	7-21
在新主机上还原和恢复数据库	7-22
准备将数据库还原到新主机	7-23
将数据库还原到新主机	7-24
执行灾难恢复	7-28
测验	7-30
小结	7-32
练习 7 概览：使用 RMAN 执行恢复	7-33

8 监视和优化 RMAN

课程目标	8-2
并行备份集	8-3
监视 RMAN 会话	8-5
监视 RMAN 作业进度	8-7
解释 RMAN 消息输出	8-9
使用 DEBUG 选项	8-10
解释 RMAN 错误堆栈	8-11
优化 RMAN	8-12
RMAN 多路复用	8-13
分配磁盘缓冲区：示例	8-14
分配磁带缓冲区	8-15
比较同步 I/O 和异步 I/O	8-16
监视 RMAN 作业的性能	8-18
异步 I/O 瓶颈	8-19
同步 I/O 瓶颈	8-20
通道优化	8-21
优化 BACKUP 命令	8-23
优化 RMAN 备份性能	8-24
设置 LARGE_POOL_SIZE	8-25
优化 RMAN 磁带流性能瓶颈问题	8-26
测验	8-28
小结	8-30
练习 8 概览：监视和优化 RMAN	8-31

9 诊断数据库

课程目标 9-2

数据恢复指导 9-3

数据故障 9-6

数据故障：示例 9-7

数据恢复指导 RMAN 命令行界面 9-8

列出数据故障 9-9

修复建议 9-11

执行修复 9-12

分类（和关闭）故障 9-13

数据恢复指导视图 9-14

最佳做法：主动检查 9-15

什么是块损坏 9-16

块损坏故障现象：ORA-01578 9-17

如何处理损坏 9-18

设置检测损坏的参数 9-19

块介质恢复 9-21

块介质恢复的先决条件 9-22

RECOVER...BLOCK 命令 9-23

自动诊断 workflow 9-24

自动诊断资料档案库 9-25

ADR 命令行工具 (ADRCI) 9-26

V\$DIAG_INFO 视图 9-27

诊断跟踪的位置 9-28

健康装款监视器：概览 9-29

手动运行健康检查：PL/SQL 示例 9-30

使用 ADRCI 实用程序查看 HM 报表 9-31

测验 9-32

小结 9-36

练习 9 概览：诊断数据库 9-37

10 使用闪回技术 I

课程目标 10-2

闪回技术 10-3

事务处理和还原 10-4

保证还原保留时间 10-5

准备数据库以进行闪回	10-6
使用闪回技术查询数据	10-8
闪回查询	10-9
闪回查询：示例	10-10
闪回版本查询	10-11
闪回版本查询：注意事项	10-12
测验	10-13
闪回表概览	10-15
闪回表	10-16
对表启用行移动	10-17
执行闪回表	10-18
闪回表：注意事项	10-20
测验	10-21
闪回事务处理查询	10-22
使用 Oracle Enterprise Manager 执行闪回事务处理查询	10-23
闪回事务处理查询：注意事项	10-24
闪回事务处理	10-25
先决条件	10-26
闪回事务处理	10-27
可能的工作流	10-28
闪回事务处理向导	10-29
选择其它回退选项	10-30
最后步骤（不使用 EM）	10-32
测验	10-33
小结	10-34
练习 10 概览：执行闪回事务处理回退	10-35

11 使用闪回技术 II

课程目标	11-2
Oracle Total Recall 概览	11-3
设置过程	11-4
Total Recall 的工作原理	11-5
Oracle Total Recall 场景	11-6
透明的方案演化	11-9
完整方案演化	11-10
限制	11-11

准则	11-12
查看闪回数据归档	11-13
测验	11-14
闪回删除和回收站	11-16
回收站	11-17
从回收站还原表	11-19
回收站：自动回收空间	11-20
回收站：手动回收空间	11-21
不使用回收站	11-22
查询回收站	11-23
测验	11-24
小结	11-25
练习 11 概览：使用闪回技术	11-26

12 执行闪回数据库

课程目标	12-2
闪回数据库	12-3
闪回数据库体系结构	12-4
配置闪回数据库	12-5
您需要执行的操作	12-6
闪回数据库：示例	12-7
闪回数据库注意事项	12-8
监视闪回数据库	12-9
使用 EM 监视闪回数据库	12-11
保证还原点	12-12
闪回数据库和保证还原点	12-13
测验	12-14
小结	12-16
练习 12 概览：使用闪回数据库	12-17

13 管理内存

课程目标	13-2
内存管理：概览	13-3
复查 Oracle DB 内存结构	13-4
缓冲区高速缓存	13-6
使用多个缓冲池	13-8

共享池	13-10
大型池	13-11
Java 池和流池	13-12
重做日志缓冲区	13-13
自动内存管理：概览	13-14
Oracle DB 内存参数	13-15
监视自动内存管理	13-16
有效使用内存：准则	13-18
库高速缓存的内存优化准则	13-20
自动共享内存管理：概览	13-22
ASMM 的工作原理	13-23
启用自动共享内存管理功能	13-24
禁用 ASMM	13-26
程序全局区 (PGA)	13-27
使用 V\$PARAMETER 视图	13-29
测验	13-30
小结	13-31
练习 13 概览：使用 AMM 纠正内存分配问题	13-32

14 管理数据库性能

课程目标	14-2
优化活动	14-3
性能规划	14-4
实例优化	14-6
性能优化方法	14-7
性能监视	14-8
性能优化数据	14-9
优化统计信息收集	14-10
统计信息首选项：概览	14-12
使用统计信息首选项	14-13
使用 Oracle Enterprise Manager 设置全局首选项	14-14
Oracle 等待事件	14-15
实例统计信息	14-16
监视会话性能	14-18
显示与会话有关的统计信息	14-19
显示与服务有关的统计信息	14-20

故障排除和优化视图	14-21
字典视图	14-22
自动工作量资料档案库	14-24
使用自动工作量资料档案库视图	14-26
Real Application Testing 概览：数据库重放	14-27
整体情况	14-28
测验	14-29
小结	14-30
练习 14 概览：监视实例性能	14-31

15 通过 SQL 优化管理性能

课程目标	15-2
SQL 优化	15-3
SQL 指导	15-4
自动 SQL 优化结果	15-5
实施自动优化建议	15-6
SQL 优化指导：概览	15-7
使用 SQL 优化指导	15-8
SQL 优化指导选项	15-9
SQL 优化指导建议	15-10
使用 SQL 优化指导：示例	15-11
重复的 SQL	15-13
SQL 访问指导：概览	15-14
典型的 SQL 访问指导会话	15-15
工作量来源	15-16
建议选项	15-17
复查建议	15-20
SQL 性能分析程序：概览	15-21
SQL 性能分析程序：使用情形	15-22
使用 SQL 性能分析程序	15-24
测验	15-25
小结	15-29
练习 15 概览：通过 SQL 优化管理性能	15-30

16 管理资源

课程目标 16-2

数据库资源管理器：概览 16-3

数据库资源管理器：概念 16-4

使用资源管理器的原因 16-5

默认维护资源管理器计划 16-7

示例：DEFAULT_PLAN 16-8

潜在工作流 16-9

指定资源计划指令 16-11

资源计划的资源分配方法 16-12

EMPHASIS 和 RATIO 的比较 16-13

活动会话池机制 16-15

设置活动会话池 16-16

指定阈值 16-18

设置空闲超时 16-19

限制数据库级别的 CPU 占用率 16-20

限制服务器级别的 CPU 占用率：实例限制 16-22

实例限制示例 16-23

监视实例限制 16-24

资源使用者组映射 16-25

激活资源计划 16-27

数据库资源管理器信息 16-28

监视资源管理器 16-29

测验 16-32

小结 16-33

练习 16 概览：使用资源管理器 16-34

17 使用调度程序自动执行任务

课程目标 17-2

简化管理任务 17-3

核心组件 17-4

基本工作流 17-5

测验 17-7

持久轻量作业 17-8

使用基于时间的或基于事件的调度 17-9

创建基于时间的作业 17-10

创建基于事件的调度	17-12
使用 Oracle Enterprise Manager 创建基于事件的调度	17-13
创建基于事件的作业	17-14
基于事件的调度	17-15
创建复杂调度	17-17
测验	17-18
使用电子邮件通知	17-19
添加和删除电子邮件通知	17-20
创建作业链	17-22
链的示例	17-24
高级调度程序概念	17-25
作业类	17-26
窗口	17-28
区分窗口内作业的优先级	17-29
创建作业数组	17-30
测验	17-32
创建文件监视器和基于事件的作业	17-33
从远程系统启用文件到达事件	17-35
调度远程数据库作业	17-36
创建远程数据库作业	17-37
调度多个目标作业	17-38
查看调度程序元数据	17-39
测验	17-41
小结	17-42
练习 17 概览：使用调度程序自动执行任务	17-43

18 管理空间

课程目标	18-2
空间管理：概览	18-3
块空间管理	18-4
行链接和行迁移	18-5
测验	18-7
段内的空闲空间管理	18-8
段的类型	18-9
分配区	18-10
分配空间	18-11

创建没有段的表	18-12
控制延迟创建段	18-13
限制和例外	18-14
其它自动功能	18-15
测验	18-16
表压缩：概览	18-17
适用于直接路径插入操作的压缩	18-18
适用于 DML 操作的 OLTP 压缩	18-20
指定表压缩	18-21
使用压缩指导	18-22
使用 DBMS_COMPRESSION 程序包	18-23
压缩表数据	18-24
预先表空间监视	18-25
阈值和解决空间问题	18-26
监视表空间的空间使用量	18-27
收缩段	18-28
收缩操作的结果	18-29
回收 ASSM 段内的空间	18-30
段指导：概览	18-31
段指导	18-32
实施建议	18-33
自动段指导	18-34
使用 EM 手动收缩段	18-35
使用 SQL 收缩段	18-36
管理可恢复的空间分配	18-37
使用可恢复的空间分配	18-38
恢复挂起的语句	18-40
哪些操作是可恢复的	18-42
测验	18-43
小结	18-44
练习 18 概览：管理存储	18-45

19 管理数据库的空间

课程目标	19-2
数据库存储	19-3
支持 4 KB 扇区磁盘	19-4

使用 4 KB 扇区磁盘	19-5
指定磁盘扇区大小	19-6
测验	19-7
移动表空间	19-10
概念：最低兼容级别	19-11
最低兼容级别	19-12
可移动表空间过程	19-13
确定平台的 Endian 格式	19-14
使用 RMAN 的 CONVERT 命令	19-16
使用 Oracle Enterprise Manager 实施可移动表空间	19-17
移动数据库	19-20
数据库移动过程：源系统转换	19-21
数据库移动过程：目标系统转换	19-22
数据库移动：注意事项	19-23
测验	19-24
小结	19-25
练习 19 概览：管理数据库的空间	19-26

20 复制数据库

课程目标	20-2
使用副本数据库	20-3
选择数据库复制方法	20-4
复制活动数据库	20-5
通过目标连接复制数据库	20-6
在具有恢复目录但没有目标连接的情况下复制数据库	20-7
在没有恢复目录或目标连接的情况下复制数据库	20-8
创建基于备份的副本数据库	20-9
创建辅助实例的初始化参数文件	20-10
为目标指定新名称	20-11
使用 SET NEWNAME 子句	20-12
SET NEWNAME 的替代变量	20-13
指定用于文件命名的参数	20-14
在 NOMOUNT 模式下启动实例	20-16
确保备份和归档重做日志文件可用	20-17
分配辅助通道	20-18
了解 RMAN 复制操作	20-19

- 指定 DUPLICATE 命令的选项 20-21
- 使用其它 DUPLICATE 命令选项 20-22
- 使用 EM 克隆数据库 20-23
- 测验 20-24
- 小结 20-25
- 练习 20 概览：复制数据库 20-26

附录 A：练习与解答

附录 B：执行表空间时间点恢复

- 课程目标 B-2
- 表空间时间点恢复 (TSPITR)：概念 B-3
- 表空间时间点恢复 (TSPITR)：术语 B-4
- 表空间时间点恢复：体系结构 B-5
- 何时使用 TSPITR B-7
- 为 TSPITR 做准备 B-8
- 确定正确的目标时间 B-9
- 确定恢复集的表空间 B-10
- 确定跨越恢复集边界的关系 B-11
- 确定将丢失的对象 B-12
- 执行基本 RMAN TSPITR B-13
- 执行完全自动执行的 TSPITR B-14
- 使用映像副本提高 TSPITR 性能 B-15
- 使用 Oracle Enterprise Manager 执行 TSPITR B-16
- RMAN TSPITR 处理 B-17
- 使用 RMAN 管理的辅助实例执行 RMAN TSPITR B-19
- 使用您自己的辅助实例执行 RMAN TSPITR B-20
- RMAN TSPITR 故障排除 B-21
- 小结 B-22

附录 C：执行用户管理的备份和恢复

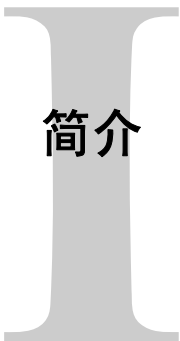
- 课程目标 C-2
- 备份和恢复的使用类型 C-3
- 执行用户管理的数据库备份 C-4
- 需要备份模式的场合 C-5
- 确定要手动备份的文件 C-6

手动备份 NOARCHIVELOG 数据库 C-7
 手动备份 ARCHIVELOG 数据库 C-8
 备份控制文件 C-9
 执行用户管理的数据库完全恢复：概览 C-10
 对关闭的数据库执行完全恢复：概览 C-11
 确定与恢复相关的文件 C-12
 还原与恢复相关的文件 C-13
 应用重做数据 C-15
 对打开的数据库执行完全恢复 C-16
 执行用户管理的不完全恢复：概览 C-17
 选择不完全恢复方法 C-18
 执行用户管理的不完全恢复 C-19
 执行用户管理的不完全恢复：步骤 C-21
 用户管理的基于时间的恢复：示例 C-22
 用户管理的基于取消的恢复：示例 C-24
 小结 C-26

附录 D：管理 ASM 实例

课程目标 D-2
 ASM 对于管理员的好处 D-3
 ASM 实例 D-4
 ASM 组件：ASM 实例 — 主进程 D-6
 ASM 实例初始化参数 D-7
 数据库实例与 ASM 之间的交互 D-9
 ASM 实例：动态性能视图 D-10
 ASM 系统权限 D-11
 使用 Oracle Enterprise Manager 管理 ASM 用户 D-12
 使用 SQL*Plus 启动和停止 ASM 实例 D-13
 使用 `srvctl` 启动和停止 ASM 实例 D-15
 使用 `asmcmd` 启动和停止 ASM 实例 D-16
 磁盘组概览 D-17
 ASM 磁盘 D-18
 分配单元 D-19
 ASM 文件 D-20
 区映射 D-21
 条带化粒度 D-22

细粒度条带化	D-23
ASM 故障组	D-25
条带化和镜像示例	D-26
故障示例	D-27
管理磁盘组	D-28
使用 SQL*Plus 创建和删除磁盘组	D-29
将磁盘添加至磁盘组	D-30
其它 ALTER 命令	D-31
使用 Oracle Enterprise Manager 的 ASM 管理	D-32
ASM 磁盘组兼容性	D-33
ASM 磁盘组属性	D-35
使用 Oracle Enterprise Manager 编辑磁盘组属性	D-36
检索 ASM 元数据	D-37
ASM 快速镜像再同步概览	D-38
小结	D-39



简介

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

学完本课后，应能完成以下工作：

- 对 Oracle DB 进行配置以实现最佳恢复
- 使用 Recovery Manager (RMAN) 备份和恢复数据库（及其部件）
- 使用闪回技术查看数据的过去状态以及将对象还原到过去状态
- 确定繁重的数据库会话和性能较差的 SQL
- 使用适当且灵活的内存配置
- 在会话和任务之间配置资源分配
- 调度作业以在数据库内部或外部运行
- 使用压缩功能来优化数据库存储并复制数据库

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

课程目标

在本课中，您将学习：

- 通过使用适当的备份和恢复策略来确保数据库的可用性
- 使用闪回技术诊断并修复数据故障
- 监视和管理主要的数据库组件，包括内存、性能和资源
- 使用调度程序自动执行 DBA 任务
- 管理空间以优化数据库存储并使之能够应对不断增长的空间需求

建议的日程表

天	课程	天	课程
1	1. Oracle DB 的核心概念和工具 2. 进行配置以实现可恢复性 3. 使用 RMAN 恢复目录 4. 配置备份设置	4	13. 管理内存 14. 管理数据库性能 15. 通过 SQL 优化管理性能 16. 管理资源 17. 使用调度程序自动执行任务
2	5. 使用 RMAN 创建备份 6. 还原和恢复任务 7. 使用 RMAN 执行恢复 8. 监视和优化 RMAN	5	18. 管理空间 19. 管理数据库的空间 20. 复制数据库
3	9. 诊断数据库 10. 使用闪回技术 I 11. 使用闪回技术 II 12. 执行闪回数据库		



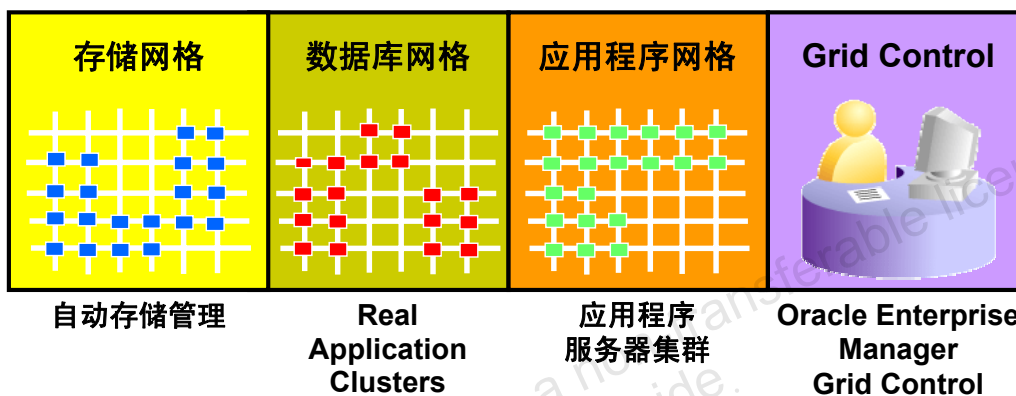
版权所有 © 2011, Oracle。保留所有权利。

建议的日程表

该日程表只是一个很笼统的大纲。您的教师将决定实际的课程日程表。

Oracle Database 11g: “g” 代表 Grid (网格)

- 开放网格论坛 (OGF)
- Oracle Grid Infrastructure:
 - 低成本
 - 高质量的服务
 - 易于管理



版权所有 © 2011, Oracle. 保留所有权利。

Oracle Database 11g: “g” 代表 Grid (网格)

开放网格论坛 (OGF) 是一个开发网格计算标准的标准化团体。这个团体由侧重制定网格计算各方面的标准的若干个委员会和工作小组构成。委员会和工作组由来自学术界、研究机构以及（日益增多的）商业公司的参与者组成。您可以参阅 OGF 的网站 <http://www.ogf.org>。

Oracle 创建的网格计算基础结构软件可用于在各个服务器间平衡所有类型的工作量，并使得所有这些服务器可作为一个完整的系统来管理。因为所有组件都是集群形式的，所以网格计算可实现与大型机计算相同的高级别可靠性。但是，与大型机和大型 UNIX 对称多处理 (SMP) 服务器不一样的是，由于在网格计算中使用了开放系统技术，如 Intel 处理器和 Linux 操作系统，所以可以以极低的成本构建网格。

Oracle 的网格计算技术包括：

- 自动存储管理 (ASM)
- Real Application Clusters (RAC)
- 应用程序服务器集群
- Oracle Enterprise Manager Grid Control

Oracle Database 11g: “g” 代表 Grid (网格) (续)

自动存储管理可以在所有磁盘之间分布数据库数据, 创建并维护存储网格, 以最低的管理成本提供最高的输入/输出 (I/O) 吞吐量。添加或删除磁盘时, ASM 将自动重新分布数据。

(不需要逻辑卷管理器来管理文件系统。) 可以使用可选的镜像功能提高数据可用性, 并且您可以联机添加或删除磁盘。请参阅“管理数据库存储结构”课程。

Oracle Real Application Clusters 在服务器集群上运行并按比例分布所有应用程序工作量, 并提供以下功能:

- **集成的集群件:** 这包括集群连接、消息传送和锁定、集群控制以及恢复功能。它可在 Oracle Database 10g 支持的所有平台上使用。
- **自动工作量管理:** 可以定义相应的规则, 以便在正常操作期间和应对故障时自动为每个服务分配处理资源。可以动态修改这些规则, 以满足不断变化的业务需求。数据库网格内的这种动态资源分配是 Oracle RAC 所特有的。
- **对中间层的自动事件通知:** 当集群配置更改时, 中间层可立即适应实例故障转移或有新实例可用等情况。这样, 最终用户在发生实例故障转移时可继续工作, 而不会发生通常情况下由网络超时导致的延迟。当有新实例可用时, 中间层可立即启动与该实例之间的负载平衡连接。Oracle Database 10g Java 数据库连接 (JDBC) 驱动程序具有“快速连接故障转移”功能, 该功能可自动启用来处理这些事件。

Oracle WebLogic Application Grid 适用于任何应用程序服务器 — 包括 Oracle WebLogic Server、IBM WebSphere Application Server 和 JBoss Application Server, 也适用于无应用程序服务器的纯网格环境。Oracle WebLogic Application Grid 提供了极致的且可预测的应用程序可伸缩性和性能。通过按需扩展容量, Oracle WebLogic Application Grid 可以将中间件基础结构从几台服务器线性扩展到数千台服务器。通过它的内存中数据网格解决方案, 可以快速访问常用的数据。利用此网格功能, 可以进行并行计算, 从而进一步改善应用程序性能。

Oracle Enterprise Manager Grid Control 管理网格级操作, 其中包括管理整个软件堆栈、预配用户、克隆数据库以及管理补丁程序。从最终用户的角度看, 它可以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, 而不是将其视为孤立的存储单元、数据库和应用程序服务器。您可以将许多硬件节点、数据库和应用程序服务器组合为单个逻辑实体, 并将一组目标作为一个单元进行管理。

注: 在本课中, 您将使用 Oracle 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 是随 Oracle Database 11g 发行版 2 (11.2) 引入的

- 是从集群件介质安装的，独立于 Oracle DB 软件
- 包含 Oracle 自动存储管理 (ASM)
- 包含 Oracle Restart — 一个适用于非集群数据库的高可用性解决方案
 - 可以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
 - ASM 实例
 - ASM 磁盘组
 - 用于 Data Guard 的 Oracle 通知服务 (ONS/eONS)

ORACLE

版权所有 © 2011, 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 是随 Oracle Database 11g 发行版 2 引入的。它是从集群件介质安装的，独立于 Oracle DB 软件，如今包括 Oracle 自动存储管理和一种称为 Oracle Restart 的新功能。

Oracle Restart 用于改进 Oracle DB 的可用性。它仅为单实例（非集群）环境实施高可用性解决方案。对于 Oracle Real Application Cluster (Oracle RAC) 环境，自动重新启动组件这一功能是由 Oracle Clusterware 提供的。Oracle Restart 可监视健康状况并自动重新启动以下组件：

- 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)

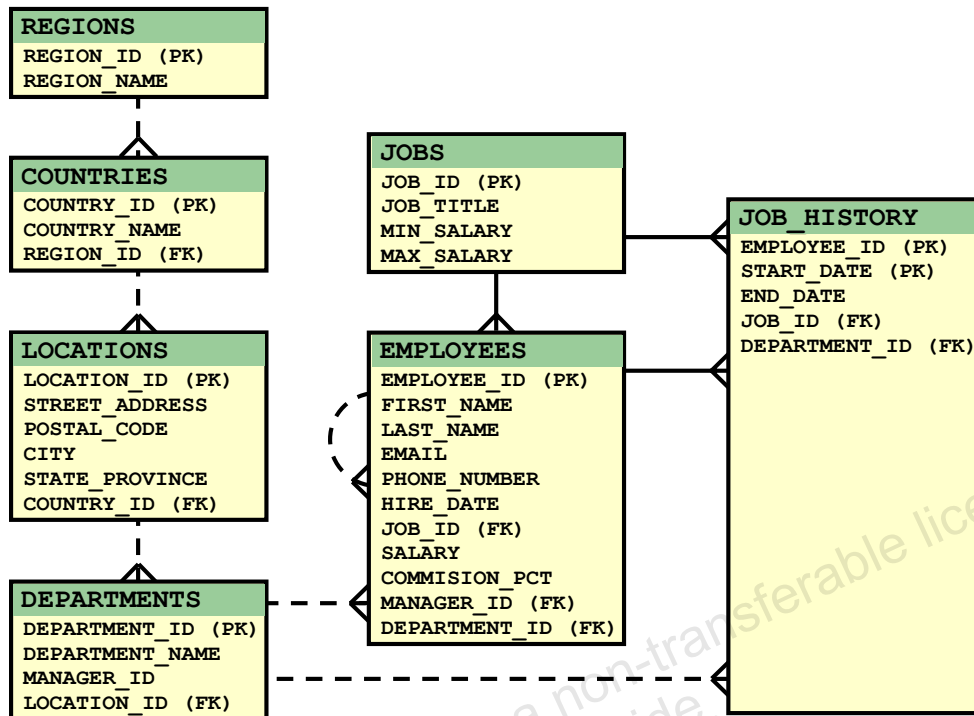
Oracle Restart 确保组件根据其依赖关系按正确的顺序启动。如果必须关闭某个组件，它可确保首先完全关闭从属组件。Oracle Restart 在与 Oracle DB 主目录分开安装的 Oracle Grid Infrastructure 主目录外运行。

用于单实例的 Oracle Grid Infrastructure（续）

一些词汇定义（方便您参考）：

- **数据库实例**是系统全局区 (SGA) 与后台进程的组合。一个实例与且只与一个数据库相关联。在 Oracle Real Application Clusters 配置中，多个实例同时访问一个数据库。
- **Oracle Net 监听程序**是一个进程，它监听传入的客户机连接请求并管理到数据库的网络通信量。
- **数据库服务**是用户创建的服务，由 Oracle Clusterware 进行管理。一种数据库服务可以在一个或多个 RAC 实例上提供，并且是基于每个实例进行管理的（在服务的启动/停止方面）。只有 Oracle Clusterware 管理的服务能够成为性能类的一部分。使用 DBMS_SERVICE 程序包创建的服务不是由 Oracle Clusterware 管理的。
- **ASM 实例**是采用与 Oracle DB 实例相同的技术构建的。ASM 实例具有与 Oracle DB 类似的系统全局区 (SGA) 和后台进程。不过，由于 ASM 执行的任务少于数据库，因此 ASM SGA 小于数据库 SGA。ASM 实例装载磁盘组，使 ASM 文件可供数据库实例使用；ASM 实例不装载数据库。
- **ASM 磁盘组**包含一个或多个 ASM 磁盘，这些磁盘是作为一个逻辑单元进行管理的。磁盘组的 I/O 自动分布在组中的所有磁盘之间。
- **Oracle 通知服务 (ONS)** 是一种发布-订阅服务，用于传输关于所有快速应用程序通知 (FAN) 事件的信息。

课程示例：HR 示例方案



ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

课程示例：HR 示例方案

本课程中使用的示例取自一个人力资源 (HR) 应用程序，可以将该应用程序创建为启动数据库的一部分。

下面是 HR 应用程序的一些主要业务规则：

- 每个部门可以是一个或多个雇员的雇主。每个雇员只可以分配到一个部门。
- 每项工作必须是一个或多个雇员的工作。每个雇员当前必须仅被分配给一项工作。
- 雇员更改其部门或工作时，JOB_HISTORY 表中的记录用于记录以前的分配的开始和结束日期。
- JOB_HISTORY 记录是由组合主键 (PK) 标识的，该组合主键包括以下列：EMPLOYEE_ID 和 START_DATE 列。

注释： PK = Primary Key（主键），FK = Foreign Key（外键）

实线表示必需的外键 (FK) 约束条件，虚线表示可选的 FK 约束条件。

EMPLOYEES 表还包含对其自身的 FK 约束条件。这是一个业务规则的实施：每个雇员只可以直接向一个经理报告。该 FK 是可选的，因为顶级雇员不向其他雇员报告。

Oracle DB 的核心概念和工具



ORACLE®

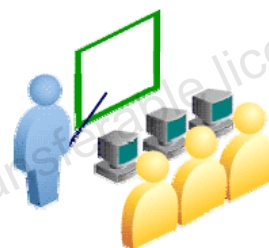
版权所有 © 2011, Oracle。保留所有权利。

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：

- 描述具有自动存储管理 (ASM) 的 Oracle DB 体系结构的核心概念
- 使用配置和管理 DBA 工具
- 描述技术课程环境



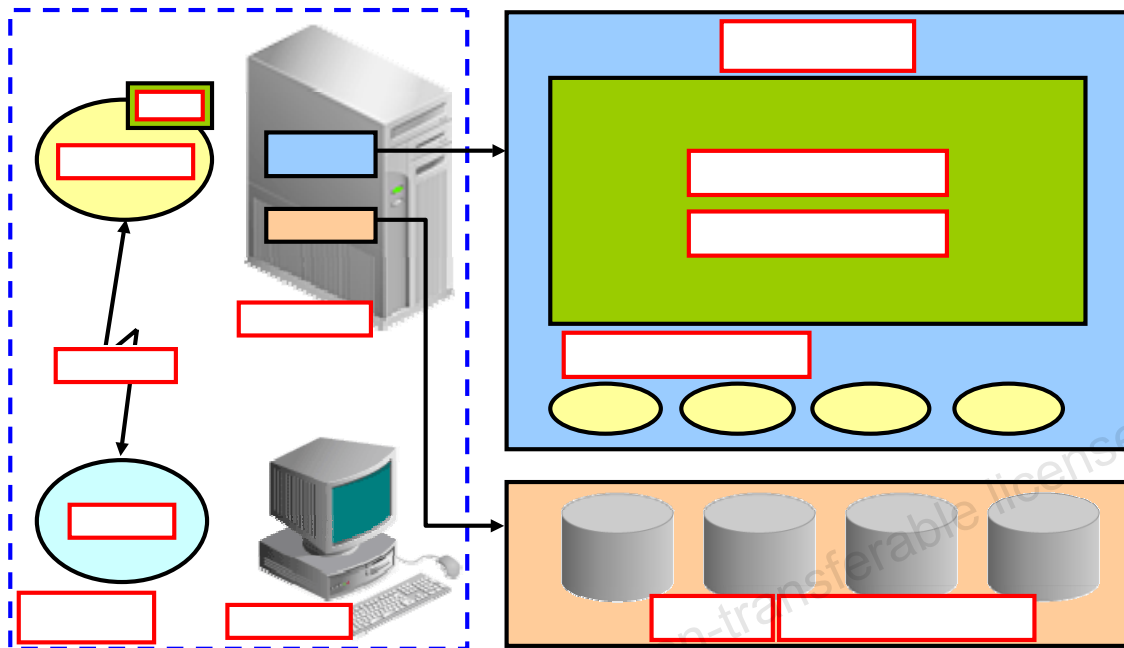
ORACLE

版权所有 © 2011, Oracle。保留所有权利。

课程目标

本课回顾了具有 ASM 的 Oracle DB 体系结构，并概述了本课程所需的技术环境。通过这种方式对核心概念和 DBA 工具进行了介绍。

列出 Oracle DB Server 的核心组件的名称



ORACLE

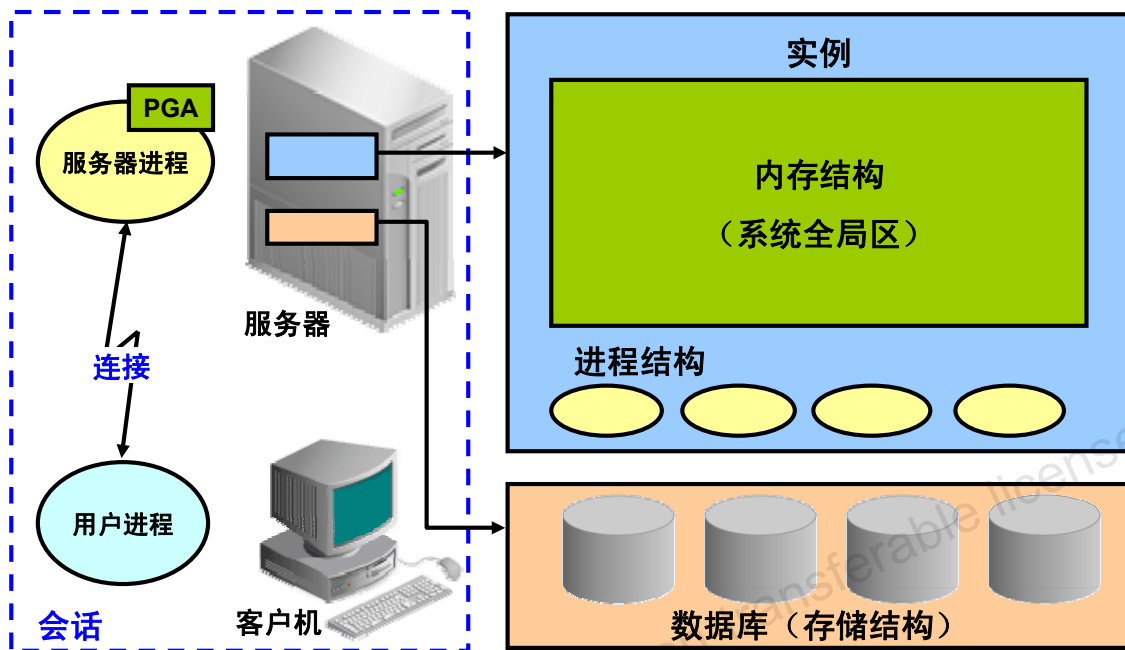
版权所有 © 2011, Oracle. 保留所有权利。

列出 Oracle DB Server 的核心组件的名称

下面是一些示例问题，可引导您通过列出核心组件的名称来获得入门知识：

1. 基本 Oracle DB 系统的两个主要组件：_____ 和 _____。
2. 实例由 _____ 和 _____ 进程组成。
3. Oracle DB Server 体系结构中的三个主要结构是：_____, _____ 和 _____。
4. 会话是 _____ 和 _____ 之间的连接。

Oracle DB Server 体系结构概览



版权所有 © 2011, Oracle. 保留所有权利。

Oracle DB Server 体系结构

Oracle DB Server 体系结构中有三个主要结构：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组成。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。启动某个数据库实例后，Oracle 软件会将该实例与特定的数据库关联。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) 针对内存和进程组件使用了实例的概念，但不与特定的数据库关联。

连接和会话都与用户进程密切相关，但意义却大不相同。

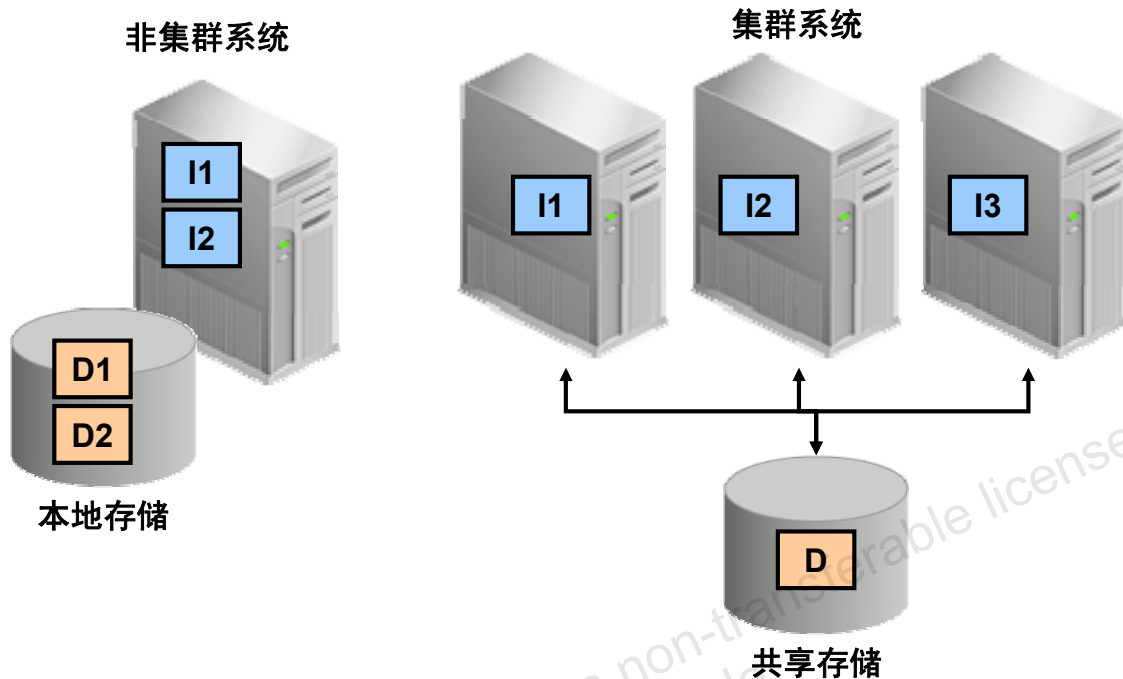
连接是用户进程和 Oracle DB 实例之间的通信路径。通信路径是使用可用的进程间通信机制（在一台同时运行用户进程和 Oracle DB 的计算机上）或网络软件（多台不同的计算机运行数据库应用程序和 Oracle DB 并通过网络进行通信时）建立的。

Oracle DB 体系结构概览（续）

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。会话从用户建立连接时开始，一直持续到用户断开连接或退出数据库应用程序。

可以使用同一个用户名为一个 Oracle DB 用户创建多个会话，并让这些会话并存。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。

实例数据库配置



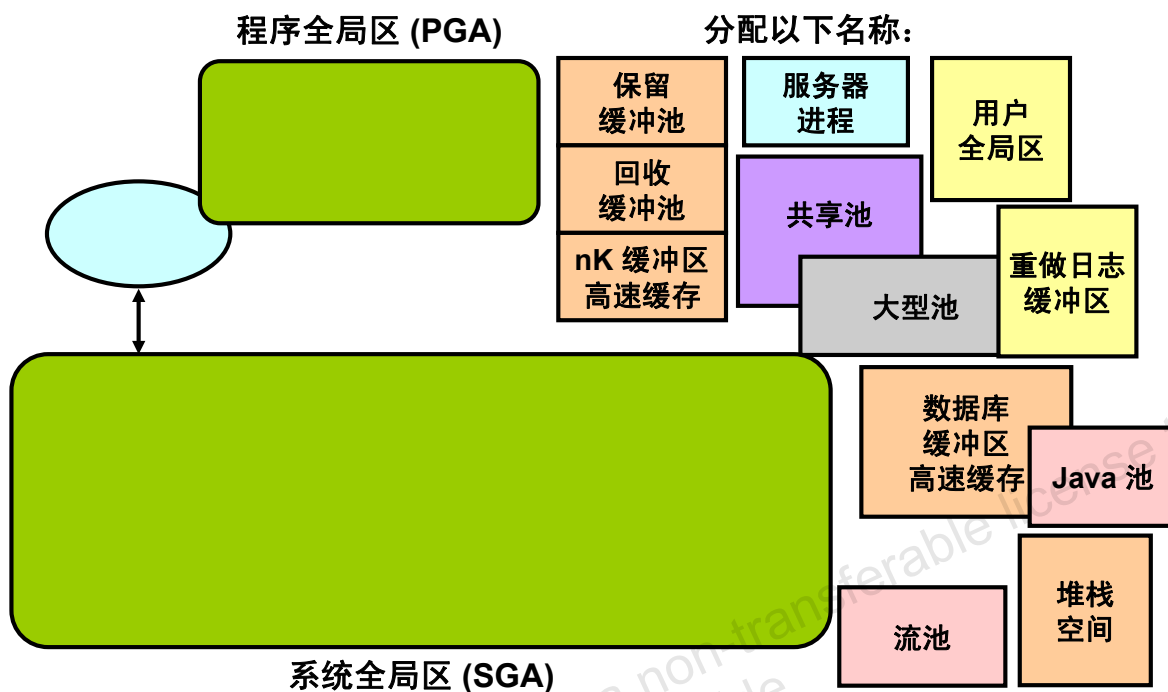
ORACLE

版权所有 © 2011, Oracle。保留所有权利。

实例数据库配置

每个数据库实例与且只与一个数据库相关联。如果同一个服务器上有多个数据库，则每个数据库具有一个单独且不同的数据库实例。数据库实例不能共享。Real Application Clusters (RAC) 数据库通常在几个独立服务器上运行同一共享数据库的多个实例。在该模型中，同一个数据库与每个 RAC 实例关联，这遵循了最多仅有一个数据库与一个实例关联这一要求。

列出 Oracle DB 的内存结构的名称



版权所有 © 2011, Oracle。保留所有权利。

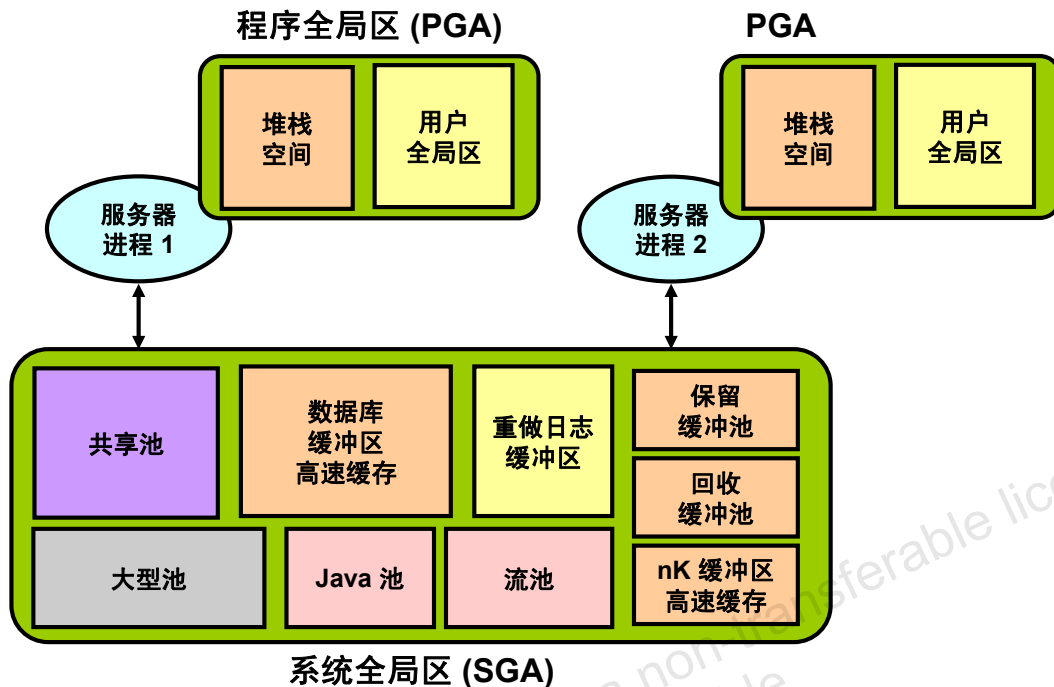
列出 Oracle DB 的内存结构的名称

入门:

1. PGA 的组件有: _____ 和 _____。
2. 列出 SGA 的主要组件的名称:

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____

Oracle DB 内存结构



版权所有 © 2011, Oracle。保留所有权利。

Oracle DB 内存结构

Oracle DB 创建并使用内存结构来满足多种用途。例如，内存可以存储正在运行的程序代码、用户间共享的数据以及每个已连接的用户的数据区域。

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 在所有服务器和后台进程之间共享。SGA 中存储的数据的示例包括高速缓存的数据块和共享 SQL 区域。
- **程序全局区 (PGA)：** 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。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 是包含实例的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
- **保留缓冲池：**一种经过优调的专用数据库缓冲区高速缓存，用以在内存中长期保留数据块
- **回收缓冲池：**一种经过优调的专用数据库缓冲区高速缓存，用以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：**用来缓存用于恢复实例的重做信息，直到可以将其写入磁盘中存储的物理重做日志文件
- **大型池：**可选区域，用于为某些大型进程（如 Oracle 备份和恢复操作）和 I/O 服务器进程分配较大的内存空间
- **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：**Oracle Streams 使用它来存储捕获和应用所需的信息

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 Server 进程为客户机请求提供服务。每个服务器进程都有自己专用的 PGA，这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 代码可对其进行读取和写入。PGA 分为两个主要区域：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，要实现此目的，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)。

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 Oracle 实例启动时启动
- 守护程序/应用程序进程
 - 网络监听程序
 - Grid Infrastructure 守护程序

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

进程体系结构

Oracle DB 系统中的进程可以分为三个主要的组：

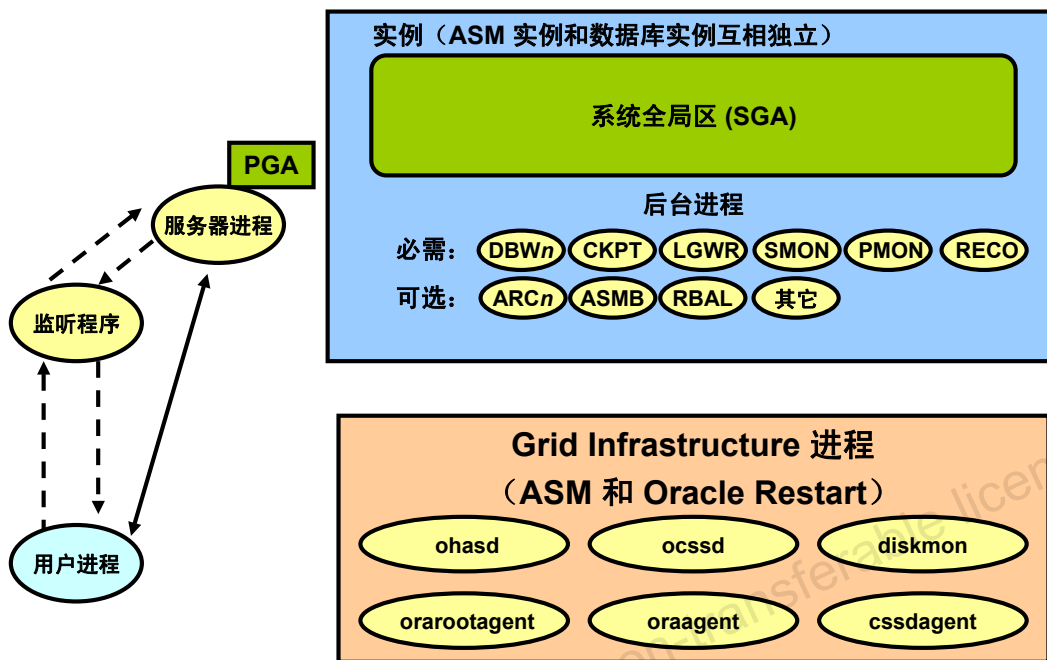
- 运行应用程序或 Oracle 工具代码的用户进程
- 运行 Oracle DB Server 代码的 Oracle DB 进程（包括服务器进程和后台进程）
- 非特定于单个数据库的 Oracle 守护程序和应用程序进程

当用户运行应用程序或 Oracle 工具（如 SQL*Plus）时，术语“用户进程”用于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 还会创建一个服务器进程来执行该用户进程发出的命令。此外，Oracle Server 还为每个实例提供一组后台进程。这些进程通过互相交互以及与操作系统进行交互来管理内存结构，通过异步执行 I/O 操作将数据写入磁盘，并执行其它必需的任务。采用不同的 Oracle DB 配置时，进程结构会随之改变，具体取决于操作系统和选择的 Oracle DB 选件。用于已连接的用户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，数据库应用程序是由一个用户进程执行的，该用户进程由执行 Oracle DB Server 代码的一个专用服务器进程提供服务。
- **共享服务器：**不需要为每个连接使用一个专用服务器进程。分派程序将传入的多个网络会话请求引向共享服务器进程池。共享服务器进程为所有客户机请求提供服务。

进程结构



版权所有 © 2011, Oracle. 保留所有权利。

进程结构

服务器进程

Oracle DB 创建服务器进程来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 位于同一台计算机上，也可以位于一台远程客户机上通过网络访问 Oracle DB。用户进程首先与一个监听程序进程通信，后者在专用环境中创建一个服务器进程。

代表每个用户应用程序创建的服务器进程可以执行以下一项或多项操作：

- 分析并运行通过应用程序发出的 SQL 语句
- 从磁盘上的数据文件中将必要的数据库块读取到 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 (DBWn)
- 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 (ARCn)
- 队列监视器进程 (QMNn)

RAC 等更高级配置可能还有其它后台进程。有关后台进程的详细信息，请参见 V\$BGPROCESS 视图。

有些后台进程是在实例启动时自动创建的，而另一些则根据需要启动。

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。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: 磁盘监视器守护程序，负责 HP Oracle Exadata Storage Server 的输入和输出防护
- cssdagent: 启动、停止和检查 CSS 守护程序 ocssd 的状态
- oraagent: 扩展集群件以支持特定于 Oracle 的要求和复杂资源
- orarootagent: 一个专用的 Oracle 代理进程，可帮助管理 root 用户所拥有的资源（如网络）

添加进程名称

- | | |
|----------------------------------|-------------------|
| 1. _____ 进程将灰数据缓冲区写入数据文件。 | A. 检查点进程 (CKPT) |
| 2. _____ 进程将重做条目写入联机重做日志文件。 | B. 系统监视器进程 (SMON) |
| 3. _____ 进程将检查点信息写入控制文件和每个数据文件头。 | C. 恢复器进程 (RECO) |
| 4. _____ 进程在实例启动时执行恢复。 | D. 日志写进程 (LGWR) |
| 5. _____ 进程在用户进程失败时执行进程恢复。 | E. 归档进程 (ARCn) |
| 6. _____ 进程解决未决分布式事务处理。 | F. 进程监视器进程 (PMON) |
| 7. _____ 进程将重做日志文件复制到指定的存储设备。 | G. 数据库写进程 (DBWn) |

ORACLE

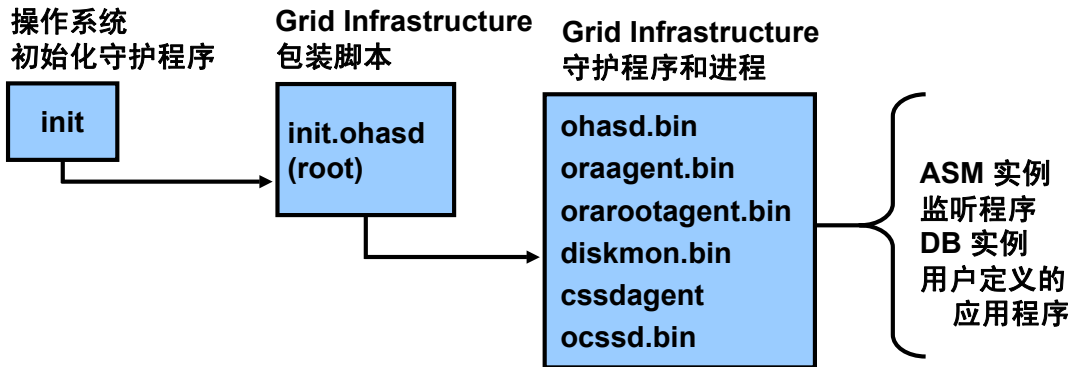
版权所有 © 2011, Oracle。保留所有权利。

添加进程名称

填写进程名称之后，可参阅“附录 A 解答”中的参考答案。

进程启动顺序

- Oracle Grid Infrastructure 由操作系统初始化守护程序启动。



- Oracle Grid Infrastructure 安装会修改 `/etc/inittab` 文件，以确保它自己在计算机每次启动时在相应运行级别启动。

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

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

进程启动顺序

安装 Oracle Grid Infrastructure 时，会将条目放置在操作系统的 `/etc/inittab` 文件中，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 中的 `/etc/inittab` 文件格式如下：

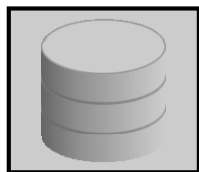
```
id : run levels : action : process with parameters
```

由于包装脚本是随 `respawn` 操作启动的，因此每次终止后，它都将重新启动。

Oracle Grid Infrastructure 守护程序启动以后，有些将以 `root` 用户身份以实时优先级运行，而另一些将以 Oracle Grid Infrastructure 所有者身份以用户模式优先级运行。在 Windows 平台上，将使用操作系统服务而不是包装初始化脚本，并且守护程序是可执行二进制文件。

注：不支持直接执行包装脚本。

数据库存储体系结构



控制文件



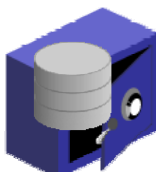
数据文件



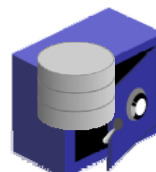
联机重做日志文件



参数文件



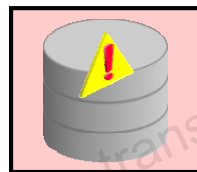
备份文件



归档重做日志文件



口令文件



预警日志和跟踪文件

ORACLE

版权所有 © 2011, 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：**用于恢复数据库实例。如果数据库服务器崩溃，但未丢失任何数据文件，则实例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：

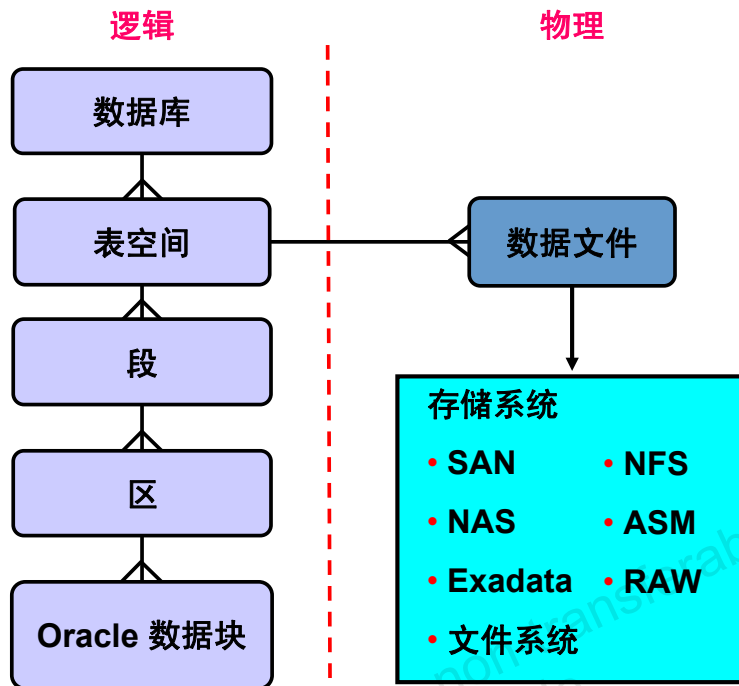
- **参数文件：**用于定义实例启动时如何对其进行配置。
- **口令文件：**允许 sysdba、sysoper 和 sysasm 远程连接到实例并执行管理任务。
- **备份文件：**用于数据库恢复。如果介质故障或用户错误导致原始文件被损坏或删除，则通常要还原备份文件。
- **归档重做日志文件：**包含实例生成的数据更改（重做）的实时历史记录。使用这些文件和数据库备份，可以恢复丢失的数据文件。也就是说，使用归档日志可以恢复还原的数据文件。

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关联的跟踪文件。当进程检测到内部错误时，进程会将有关该错误的信息转储到相应的跟踪文件中。写入跟踪文件的部分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按时间顺序列出的消息日志和错误日志。Oracle 建议您定期查看预警日志。

注：参数、口令、预警和跟踪文件在其它课程中阐述。

逻辑和物理数据库结构



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

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。系统为每个表空间显式创建一个或多个数据文件，以便在物理上将所有段的数据都存储在表空间中。如果表空间是一个 TEMPORARY 表空间，则这个表空间有一个临时文件，而不是数据文件。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

一个数据库分为多个逻辑存储单元，这些单元称为“表空间”，用于将相关逻辑结构或数据文件归到一组。例如，表空间通常会将应用程序的所有段归到一组，以简化一些管理操作。

数据块

从最细的层面来讲，Oracle DB 的数据存储在数据块中。一个数据块与磁盘中特定字节数的物理空间相对应。每个表空间的数据块大小是在创建表空间时指定的。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。一个区中的 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

区的上一级逻辑数据库存储称为“段”。段是为某个逻辑结构分配的一组区。例如：

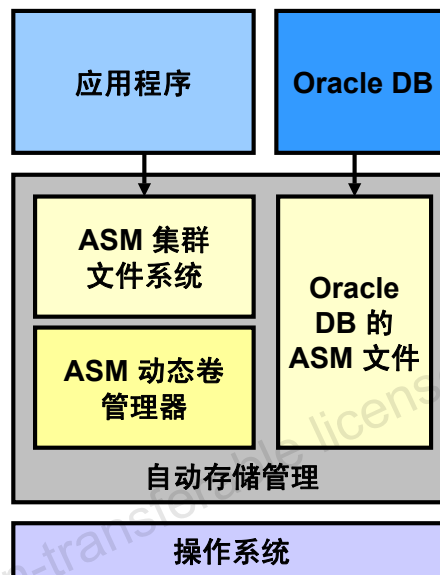
- **数据段：**每一个以非集群、非索引方式组织的表都有一个数据段，但外部表、全局临时表和分区表除外，它们每一个都有一个或多个段。表中的所有数据都存储在相应数据段的区中。对于分区表，每个分区都有一个数据段。每个集群也都有一个数据段。集群中每个表的数据都存储在集群的数据段中。
- **索引段：**每个索引都有一个索引段，存储其所有数据。对于分区索引，每个分区都有一个索引段。
- **还原段：**针对每个数据库实例，都会创建一个 UNDO 表空间。该表空间包含大量用于临时存储还原信息的还原段。还原段中的信息用于生成读一致性数据库信息，并且在数据库恢复过程中，用于为用户回退未提交的事务处理。
- **临时段：**临时段是在需要临时工作区来执行 SQL 语句时由 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。可以为每个用户指定一个默认临时表空间，也可以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。也有一些方案对象（如视图、程序包、触发器，等等）虽然是数据库对象，但不被视为段。段拥有单独的磁盘空间分配。其余对象存储为系统元数据段中的行。

Oracle DB 对空间进行动态分配。如果段中的现有区都已满，将会再增加一些区。因为区是根据需要分配的，因此段中的区在磁盘中可能是相邻的，也可能是不相邻的。这些区可以来自属于同一个表空间的不同数据文件。

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
- 对数据进行镜像以防范故障
- 解决存储管理难题



ORACLE

版权所有 © 2011, 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的纵向集成。ASM 可以管理单个对称多处理 (SMP) 计算机，也可以管理集群的多个节点以支持 Oracle Real Application Clusters (RAC)。

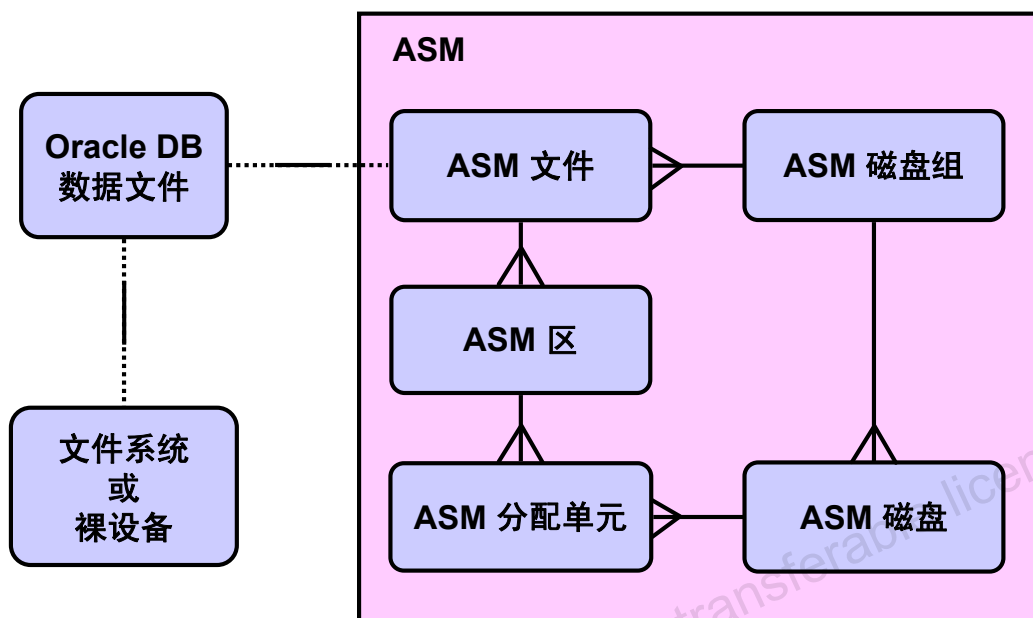
Oracle ASM 集群文件系统 (ACFS) 是一种多平台、可伸缩的文件系统和存储管理技术，该技术扩展了 ASM 的功能，可支持 Oracle DB 外部的应用程序文件，如可执行文件、报表、BFILE、视频、音频、文本、图像以及其它一般用途的应用程序文件数据。

ASM 将输入/输出 (I/O) 负载分布在所有可用的资源中以优化性能，不必手动进行 I/O 优化。ASM 帮助 DBA 管理动态数据库环境，使 DBA 不必关闭数据库，就可以增加数据库的大小来调节存储分配。

ASM 可以通过维护冗余的数据副本来提供容错能力，另外，也可以基于供应商提供的存储机制来构建 ASM。数据管理将通过为各类数据选择所需的可靠性和性能指标来实现的，而不是人工处理每个文件。

通过使原本手动执行的存储工作自动化，ASM 功能节省了 DBA 的时间，从而提高了管理员的能力，使其可以更为高效地管理更多和更大的数据库。

ASM 存储组件



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

ASM 存储组件

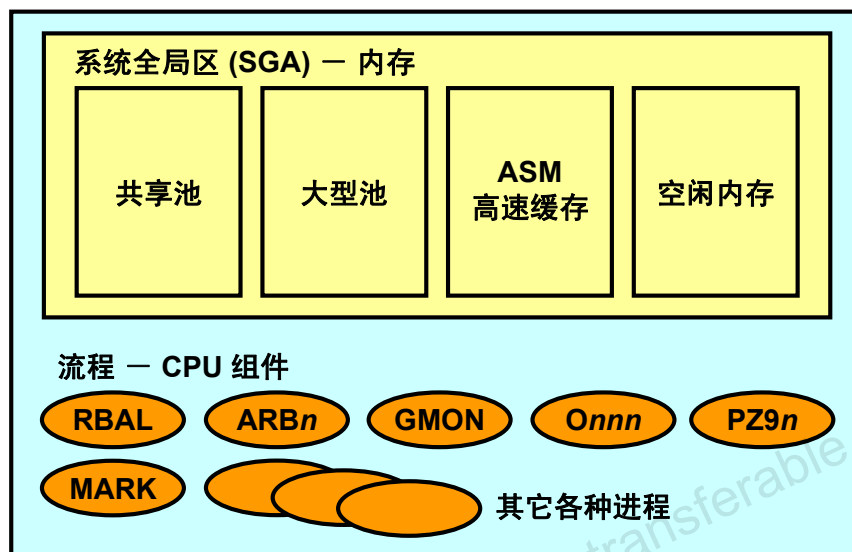
ASM 不会妨碍任何现有的数据库功能。现有数据库能够像平常一样工作。新文件可以被创建为 ASM 文件，而现有文件既可以按原来的方式进行管理，也可以迁移到 ASM。

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 数据文件与文件系统中存储在操作系统上的文件具有一对一关系，或者与 ASM 文件具有一对一关系。

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) 或附加到网络的文件。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 分配单元形成一个 ASM 区。Oracle ASM 区是用于存放 Oracle ASM 文件内容的裸存储。一个 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。

ASM 实例

ASM 实例是 ASM 的进程和内存组件。



ASM 实例

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

ASM 实例

每次启动 ASM 或数据库时，都会分配一个称为系统全局区 (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，如下所述：

- **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

建议为 ASM 实例使用的最低内存量为 256 MB。自动内存管理在 ASM 实例上默认启用，它将动态优化各个 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。一个 ASM 实例可以具有许多后台进程；并不是总是会出现所有进程。

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述：

- **ARCn**: 归档进程
- **CKPT**: 检查点进程
- **DBWn**: 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**: 进程衍生程序进程
- **QMNn**: 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，这些进程不是始终会执行在数据库实例中会执行的任务。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 进程将日志记录信息复制到 ASM 磁盘组。

如果 ASM 是集群形式的，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述：

- **LMON**: 全局入队服务监视器进程
- **LMDn**: 全局入队服务守护程序
- **LMSn**: 全局高速缓存服务进程
- **LCKn**: 锁进程

有关管理 ASM 实例的更多信息，请参阅附录 D 和文档。

DBA 配置工具

为本课程设置技术环境涉及以下任务和工具：

- 使用 OUI 为独立服务器安装和配置 Oracle Grid Infrastructure，包括：
 - 配置监听程序
 - 创建 ASM 实例 (+ASM) 和配置 +DATA 磁盘组
 - 配置 Oracle Restart
- 使用 asmca 创建和配置其它 ASM 磁盘组（如 +FRA）
- 使用 OUI 安装 Oracle Database 11g 软件
- 使用 dbca 创建 orcl Oracle DB

注：已经替您执行了这些任务。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

DBA 配置工具

有许多配置工具可用于帮助 DBA 配置和管理 Oracle 环境。其中一些工具包括：

- **Oracle Universal Installer (OUI)：**安装 Oracle 软件和选件。根据在安装的产品和安装过程中指定的选项，OUI 可以自动调用其它工具来执行其它任务，如创建 Oracle DB。
- **Oracle ASM Configuration Assistant (ASMCA)：**支持安装和配置 ASM 实例、磁盘组、卷和 Oracle ASM 集群文件系统 (Oracle Automatic Storage Management Cluster File System, Oracle ACFS)。
- **Oracle Database Configuration Assistant (DBCA)：**允许您创建和删除 Oracle DB，以及修改数据库选项和管理数据库模板。如果在安装过程中选择创建数据库，则在安装 Oracle DB 软件的过程中会从 OUI 调用该实用程序。
- **Net Configuration Assistant (NETCA)：**配置监听程序和命名方法，它们是 Oracle 环境的重要组成部分。

注：对于独立服务器安装，Oracle Restart 是作为 Oracle Grid Infrastructure 的一部分安装和配置的。Oracle Restart 管理 Oracle 组件依赖关系，并且在硬件或软件发生故障之后或者数据库主机重新启动时自动重新启动各种 Oracle 组件。如果在安装 Oracle DB 软件之前已安装了 Oracle Grid Infrastructure，则将自动使用 Oracle Restart 配置所创建的任何数据库。

管理框架和相关 DBA 工具

Oracle DB 管理框架包括：

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

相关工具和命令包括：

SQL*Plus:

```
SQL> startup
SQL> shutdown immediate
```

监听程序控制实用程序：

```
$ lsnrctl status
```

Oracle Enterprise Manager 控制实用程序：

```
$ emctl status dbconsole
$ emctl start dbconsole
$ emctl stop dbconsole
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理框架和相关 DBA 工具

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。这可以是在运行数据库服务器的节点上运行的管理代理（管理代理将数据库服务器连接到 Oracle Enterprise Manager Grid Control），也可以是独立的 Oracle Enterprise Manager Database Control。该界面也称为“数据库控制台”。

用于启动、停止或检查状态的相关工具和命令有：

- 对于数据库实例，在 SQL*Plus 中有：startup 和 shutdown immediate。
- 对于监听程序：lsnrctl status。
- 对于 Oracle Enterprise Manager Database Control:
 - emctl dbconsole status
 - emctl dbconsole start
 - emctl dbconsole stop

使用 Oracle Restart 简化数据库管理

- 在重新启动主机时或者发生硬件或软件故障后重新启动 Oracle 组件
- 监视组件并在需要时重新启动它们
- 适用于单实例环境
- 考虑组件依赖关系：
 - 在启动数据库实例之前装载磁盘组并启动 ASM 实例
 - 数据库实例和监听程序之间的软依赖关系
- 使用 `crsctl` 实用程序启动 Oracle Restart
- 使用 `srvctl` 实用程序管理 Oracle Restart 组件

```
$ srvctl stop database -d orcl -o abort
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 Oracle Restart 简化数据库管理

- 使用 Oracle Restart 时，在发生硬件或软件故障后或者在数据库主机重新启动时，各种 Oracle 组件可自动重新启动。
- Oracle Restart 执行定期检查来监视这些组件的健康状况。如果对某个组件的检查操作失败，则将关闭并重新启动该组件。
- Oracle Restart 仅用于单实例（非集群）环境。对于 Oracle Real Application Clusters (Oracle RAC) 环境，自动重新启动组件这一功能是由 Oracle Clusterware 提供的。
- Oracle Restart 考虑 Oracle 组件的依赖关系，确保它们以合适的顺序启动。例如，如果数据库文件存储在 ASM 磁盘组中，则在启动数据库实例之前，Oracle Restart 会确保 ASM 实例已启动，并且所需的磁盘组已装载。同样，如果必须关闭某个组件，则 Oracle Restart 会确保首先完全关闭从属组件。
- Oracle Restart 还管理数据库实例与 Oracle Net 监听程序（监听程序）之间的软依赖关系。当数据库实例启动时，Oracle Restart 会尝试启动监听程序。如果监听程序启动失败，数据库仍将启动。如果监听程序稍后发生故障，Oracle Restart 不会关闭，也不会重新启动任何数据库实例。

使用 Oracle Restart 简化数据库管理（续）

- 可使用 Clusterware Control (crsctl) 实用程序启动 Oracle Restart。
- Oracle Restart 包括 Server Control (srvctl) 实用程序，该实用程序用于启动和停止 Oracle Restart 管理的组件。

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，应当使用 Oracle DB 软件中的 srvctl 实用程序。启动 ASM 实例或监听程序时，应当使用 Grid Infrastructure 软件中的 srvctl 实用程序。

测验

对于独立服务器安装，Oracle Restart 是作为 Oracle Grid Infrastructure 的一部分安装和配置的。

1. 对
2. 错

ORACLE

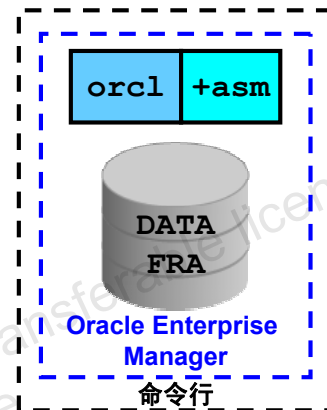
版权所有 © 2011, Oracle。保留所有权利。

答案：1

小结

在本课中，您应该已经学会：

- 描述具有 ASM 的 Oracle DB 体系结构的核心概念
- 确定哪些 DBA 配置和管理工具用于哪些任务
- 描述技术课程环境



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

小结

- 幻灯片中的图表将 orcl 和 +asm 实例显示为数据库圆柱体上方的矩形，该数据库圆柱体列出了 DATA 和 FRA 磁盘组。
- 它们由蓝色虚线围绕，将 Oracle Enterprise Manager 表示为一个可能的图形界面。
- 所有这些又被黑色虚线围绕，将命令行表示为可能的基于字符的界面。

2 进行配置以实现可恢复性

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

学完本课后，应能完成以下工作：

- 调用和配置 Recovery Manager (RMAN)
- 在 ARCHIVELOG 模式下配置数据库
- 配置多个归档日志文件目标以提高可用性
- 配置快速恢复区 (FRA)
- 指定保留策略

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

课程目标

快速恢复区 (Fast Recovery Area, FRA) 以前称为 Flash Recovery Area。

备份和恢复功能的用途

下列任务需要备份和恢复功能：

- 数据保护
 - 介质故障
 - 用户错误
 - 应用程序错误
- 数据保留和历史保留
- 数据传输

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

备份和恢复功能的用途

备份和恢复的用途是还原失败的数据库。备份用于在出现以下问题时保护数据库：硬件故障、介质故障、用户错误和应用程序错误。介质错误会因硬件级故障而引发数据问题；控制器故障或磁盘驱动器故障可能导致不太明显的错误，也可能导致明显的错误。如果用户发出不该发出的命令，则还会导致数据错误。应用程序故障也会导致这些相同类型的错误。

备份还可用于数据保留和历史保留。在 ARCHIVELOG 模式下创建和保留的备份可用于将数据库恢复到过去的某个时间点（该时间点用于满足法规要求）。

您还可以使用备份和恢复工具将数据移到其它数据库，甚至移到其它位置。数据库备份是将数据库复制到其它位置的一种可能方法。

典型的备份和恢复任务

为了能够以最短的停机时间从数据丢失问题中恢复，请执行以下任务：

- 配置数据库以实现可恢复性
- 定义备份计划
- 对各种类型的故障情形进行计划和测试
- 监视、优化备份和恢复环境，并排除其中的问题
- 从备份还原数据
- 将事务处理恢复到所需的时间点

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

典型的备份和恢复任务

如果数据库损坏会造成非常严重的后果，则制订一个可靠的备份和恢复计划非常重要。该计划包含下列任务：

- **配置：**需要针对您的环境配置备份和恢复策略。这应该包括备份方法、备份目标、备份保留时间和备份的删除，必要时还应包括备份保护（加密）。
- **日程安排：**应该将备份安排在非峰值时段自动运行。
- **测试：**定期测试备份和恢复活动。
- **监视：**监视备份操作的影响以确定生产数据库上的性能下降，并在必要时改进备份效率。
- **还原：**用数据文件的备份覆盖损坏的数据文件。数据文件在当前数据库之前的一个时间点。
- **恢复：**恢复使用归档和重做信息将更改应用于各个块，将数据库向前移动至当前时间点。

Oracle 备份和恢复解决方案

对于可恢复系统：

- RMAN
 - 块介质恢复
 - 压缩未使用的块
 - 二进制压缩
 - 备份加密
- 解决方案是通过以下备份类型实现的：
 - 所选文件中的所有数据块（完全备份或增量级别为 0 的备份）
 - 只限自以前某次备份以来更改过的信息（增量备份）
 - 累积（自上次 0 级备份以来的更改）
 - 差异（自上次增量备份以来的更改）

0
1

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Oracle 备份和恢复解决方案

下面是主要的备份和恢复解决方案：

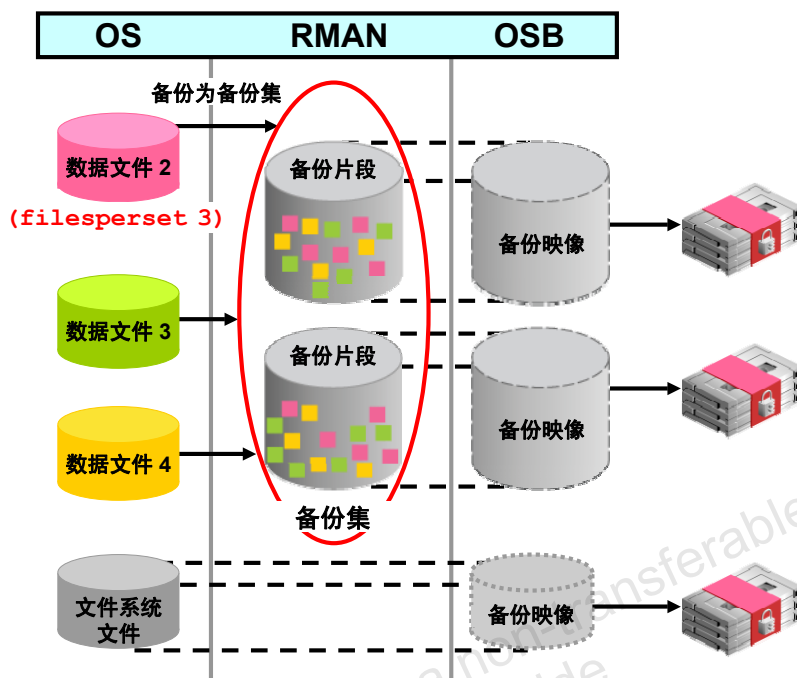
Recovery Manager: 用于执行备份和恢复的实用程序（具有图形界面和命令行界面）。使用 RMAN 时，可使用的一些主要功能有：

- **块介质恢复:** 一种恢复特定数据块的方法，与恢复整个表（通过数据泵）或数据文件（通过 RMAN）相对。
- **压缩未使用的块:** 通过压缩未使用的块这一方法，备份将不读取数据库当前未使用的块，因此，备份中将不包括未使用的块。
- **二进制压缩:** 一种节省空间的功能，它使用大家熟知的算法（与 Linux 中的 zip 等实用程序类似）压缩备份文件。
- **备份加密:** 一种用于保护所创建备份的安全设备。

解决方案是通过以下备份类型实现的：

- **完全备份:** 为包含数据且位于要备份的文件中的每个数据块创建副本。
- **增量备份:** 为自以前某次备份以来发生更改的所有数据块创建副本。Oracle DB 支持两个级别的增量备份（0 和 1）。1 级增量备份可以为以下两种类型之一：累积或差异。累积备份用于备份自上次 0 级备份以来发生的所有更改。差异备份用于备份自上次增量备份（可以为 0 级或 1 级备份）以来发生的所有更改。

Oracle 备份解决方案



版权所有 © 2011, Oracle。保留所有权利。

Oracle 备份解决方案

幻灯片显示了备份集。左侧部分是 OS 级的数据文件。可看到这些文件与 RMAN 映像副本和备份片段（中间的段）的关联方式以及这些文件与 Oracle Secure Backup (OSB) 备份映像的关联方式。

图表的底部是文件系统文件，这些文件没有 RMAN 等效体；它们直接关联到 OSB 备份映像。

- RMAN 备份数据文件、控制文件、重做日志归档和 SPFILE，无论这些文件是原始文件、备份集还是映像副本。RMAN 执行以磁盘为目标的备份和恢复操作，在介质管理层 (MML)（如 Oracle Secure Backup）的帮助下还可以备份至磁带或从磁带备份。
- Oracle Secure Backup (OSB) 是用于整个 Oracle 环境（包括文件系统和 Oracle DB）的集中式磁带管理软件。OSB 可以在本地备份和还原数据，也可以通过局域网 (LAN)、广域网 (WAN) 或 SAN 备份和还原数据。

该图表显示了 Oracle 备份解决方案的一个子集。其中未包括一些媒体管理器可能支持的映像副本和代理副本。

术语回顾

请将以下术语和说明进行匹配：

1. ____ 备份数据库的一部分。其中可能包括也可能不包括控制文件。
2. ____ 是一致备份，因为数据文件头中的 SCN 与控制文件中的 SCN 匹配。
3. ____ 为包含数据且位于要备份的文件中的每个数据块创建备份。
4. ____ 是非一致备份，因为不能确保数据文件与控制文件同步。
5. ____ 包括所有数据文件和至少一个控制文件。

(W) 整个数据库备份 (F) 完全备份 (C) 冷备份或脱机备份

(P) 部分数据库备份 (O) 联机备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

术语回顾

正确答案：1P、2C、3F、4O、5W

术语回顾

哪个说明最符合以下备份类型？

1. 映像副本
2. 备份集

说明：

- A) 它们是一个或多个二进制文件的集合，这些二进制文件包含一个或多个数据文件、控制文件、服务器参数文件或归档日志文件。不存储空数据块和当前未使用的块。
- B) 它们是数据文件或归档日志文件的副本（类似于文件副本）。

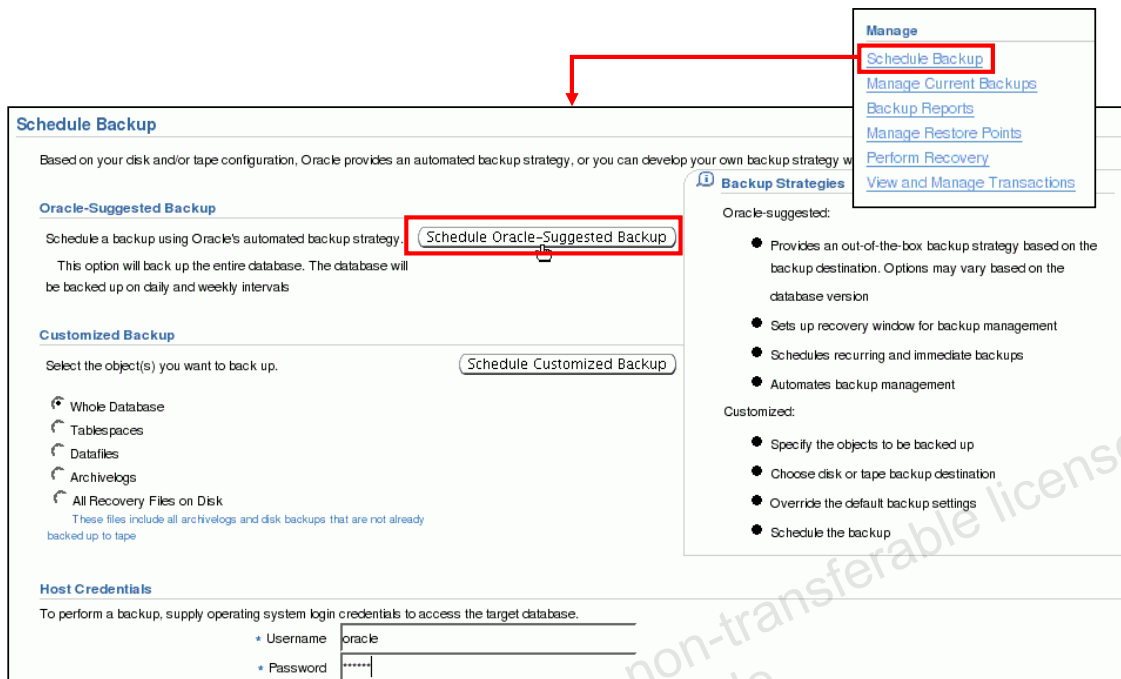
ORACLE

版权所有 © 2011, Oracle。保留所有权利。

术语回顾（续）

正确答案：1B、2A

您已了解的内容：Oracle 建议的备份



版权所有 © 2011, Oracle。保留所有权利。

您已了解的内容：Oracle 建议的备份

使用 Oracle Enterprise Manager 可以很容易地设置 Oracle 建议的备份策略，该策略可以保护数据并提供有效的恢复，可恢复到过去 24 内的任意时间点，甚至还有可能恢复到 48 小时前，具体取决于上次备份创建的时间。Oracle 建议的策略使用了增量备份和增量更新功能，与从归档日志文件应用数据库更改相比，可提供更快的恢复速度。

要建立 Oracle 建议的策略，请导航到“Maintenance（维护）”页。在“Backup/Recovery（备份/恢复）”区域中，选择“Schedule Backup（调度备份）”。使用“Backup Strategies（备份策略）”部分可在“Oracle-Suggested Backup（Oracle 建议的备份）”策略和“Customized Backup（定制备份）”策略中进行选择。Oracle 建议的策略是在第一次备份时采用完全数据库复制。由于这是整体数据库备份，您可能要在活动最少的时段进行此备份。之后，每天执行到磁盘的增量备份。也可以选择每周执行磁带备份，以备份与恢复有关的所有文件。

因为保留了磁盘上的备份，所以始终可执行完全数据库恢复，或者至少可以执行到过去 24 小时内任意时间的时间点恢复。恢复时间最长可达过去 48 小时。这是因为在指定日期创建备份之前，从第 $n-1$ 天起的备份仍存在。

使用 Recovery Manager

```

$ rman target /

RMAN> BACKUP DATABASE;
Starting backup at 10-JUN-07
.
.
RMAN> LIST BACKUP;
BS Key   Type LV Size       Device Type Elapsed Time Completion Time
-----
1        Full  1.06G    DISK          00:01:49      10-JUN-07
.
.
RMAN> DELETE OBSOLETE;
.
.
Do you really want to delete the above objects (enter YES or NO)? YES
deleted archived log
.
.

```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 Recovery Manager

在操作系统命令行中调用 RMAN，并指定相应选项。下面是最常用的选项：

- **target:** 目标数据库的连接字符串
- **catalog:** 恢复目录的连接字符串
- **nocatalog:** 指定不存在恢复目录。这是默认设置
- **cmdfile:** 输入命令文件的名称
- **log:** 输出消息日志文件的名称

幻灯片中显示的 RMAN 调用只连接到作为目标的本地数据库。

下面是一个 RMAN 调用示例，该示例使用 OS 验证连接到本地数据库，并指定要运行的命令文件和要接收属于该会话的 RMAN 命令副本的日志文件：

```
$ rman target / cmdfile=~/fullbu.rman log=~/fullbu.log
```

在 RMAN 提示符下，可以通过提交 RMAN 命令来根据需求以多种不同的方式管理备份环境和创建备份。幻灯片中显示的命令用于列出现有的备份 (LIST BACKUP)，删除任何过时的备份 (DELETE OBSOLETE)。本课程不会具体介绍这些命令和其它命令。

注：有关如何调用 RMAN 的详细信息，请参阅《Oracle Database Backup and Recovery User's Guide》。有关 RMAN 命令及其选项的完整列表，请参阅《Oracle Database Backup and Recovery Reference》。

RMAN 命令的类型

RMAN 命令具有以下类型：

- 独立命令：
 - 在 RMAN 提示符下单独执行
 - 不能在 RUN 内作为子命令出现
- 作业命令：
 - 必须位于 RUN 命令的大括号内
 - 作为一个组执行

有些命令既可以作为独立命令执行，也可以作为作业命令执行。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

RMAN 命令的类型

您可以发出两种基本类型的 RMAN 命令：独立命令和作业命令。

独立命令在 RMAN 提示符下执行，通常是自包含的。以下是一些独立命令：

- CHANGE
- CONNECT
- CREATE CATALOG、RESYNC CATALOG
- CREATE SCRIPT、DELETE SCRIPT、REPLACE SCRIPT

作业命令通常被组合在命令块内并按顺序执行。如果块内的任一命令失败，RMAN 就会停止处理，不再执行块内的其它命令。但仍然保留所有已执行命令的结果，不会以任何形式撤消。

只能作为作业命令运行的命令的示例有 ALLOCATE CHANNEL。因为仅为执行作业分配通道，所以该命令不能作为独立命令发出。有些命令既可通过提示符发出，又可在 RUN 命令块内发出，如 BACKUP DATABASE。如果发出的是独立命令，则 RMAN 将使用自动通道分配功能分配所需的通道。

您可以交互模式或批处理模式执行独立命令和作业命令。

作业命令：示例

作业命令出现在 RUN 命令块内：

```
RMAN> RUN
2> {
3>   ALLOCATE CHANNEL c1 DEVICE TYPE DISK
4>     FORMAT "/disk2/%U";
5>   BACKUP AS BACKUPSET DATABASE;
6>   SQL 'alter system archive log current';
7> }
```

当输入此行时，将开始执行整个块

在 RUN 块完成后取消分配

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

作业命令：示例

与独立命令不同，作业命令必须出现在 RUN 命令的大括号内。幻灯片中所示的 RUN 块内的命令是作为单个命令单元运行的。在 RUN 块中进行的任何配置都应用于整个块，并覆盖以前的所有设置。下面是作业命令的示例，这些命令必须位于 RUN 块内部：

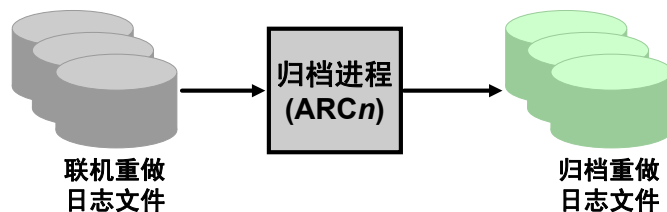
- ALLOCATE CHANNEL
- SWITCH

RMAN 按顺序执行 RUN 命令块内的作业命令。如果块内的任一命令失败，RMAN 就会停止处理，不再执行块内的其它命令。事实上，RUN 命令定义了一个命令执行单元。当 RUN 块内的最后一个命令完成后，Oracle DB 会释放所有服务器端资源，如块内分配的输入/输出 (I/O) 缓冲区或 I/O 从属进程。

注：第 6 行上的 SQL 命令只是一个示例。并不是备份操作所需的命令。

配置数据库进行备份和恢复操作

- 在 ARCHIVELOG 模式下运行数据库。



- 配置 FRA。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置数据库进行备份和恢复操作

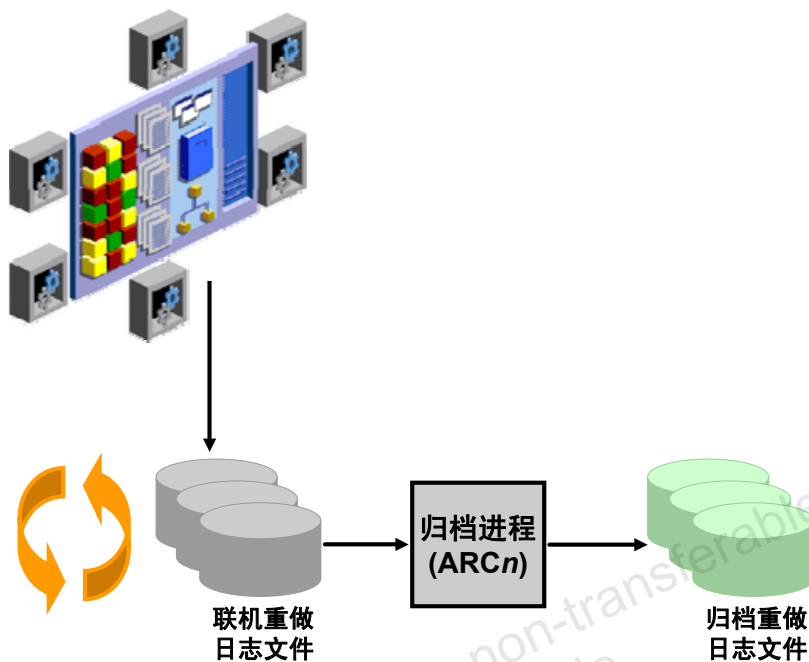
在 ARCHIVELOG 模式下运行数据库时，数据丢失后可使用更多的恢复选项，其中包括数据库或某些表空间的时间点恢复。

建议您利用快速恢复区存储尽可能多的与备份和恢复相关的文件，包括磁盘备份和归档重做日志。

有些 Oracle DB 备份和恢复功能（如 Oracle 闪回数据库和可靠还原点）需要使用快速恢复区。

本课程稍后会详细介绍这两种功能。

ARCHIVELOG 模式



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

ARCHIVELOG 模式

当对数据库中的数据进行修改后，重做数据会写出到联机重做日志文件中。指定系统在给定时间写入到的文件。当此文件写满后，归档进程 (ARCn) 会将该联机日志文件复制到其他位置，作为该文件的归档，保留时间由您决定。这便提供了更多的恢复机会，因为您可以保存、备份和还原生成的所有归档重做日志。

因为系统以循环方式重用联机重做日志文件，所以有一个协议用于控制何时可以重用文件。在 ARCHIVELOG 模式下，数据库只在联机重做日志文件归档后向其写入数据。这样可确保每个重做日志文件都有机会得以归档。

配置 ARCHIVELOG 模式

要将数据库置于 ARCHIVELOG 模式，请执行下列步骤：

- 使用 Oracle Enterprise Manager
 - 选中 “ARCHIVELOG Mode (ARCHIVELOG 模式)” 复选框。
 - 单击 “Apply (应用)” 。只有处于 MOUNT 状态时才能将数据库设置为 ARCHIVELOG。
 - 系统询问您是否重新启动数据库时，请单击 “Yes (是)” 。
- 使用 SQL 命令
 - 装载数据库。
 - 发出 ALTER DATABASE ARCHIVELOG 命令。
 - 打开数据库。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置 ARCHIVELOG 模式

将数据库置于 ARCHIVELOG 模式可防止重做日志在归档之前被覆盖。

在 Oracle Enterprise Manager 中，通过导航到 “Availability > Recovery Settings (可用性 > 恢复设置)” 并选中 “ARCHIVELOG Mode (ARCHIVELOG 模式)” 复选框可实现上述目的。在进行此更改后，必须重新启动数据库。

要发出 SQL 命令将数据库置于 ARCHIVELOG 模式，数据库必须处于 MOUNT 模式。如果数据库当前处于打开状态，必须将其完全关闭（不是中止），然后进行装载。下面显示的命令可以关闭打开的数据库，将其置于 ARCHIVELOG 模式，然后再将其打开：

```
SQL> SHUTDOWN IMMEDIATE
SQL> STARTUP MOUNT
SQL> ALTER DATABASE ARCHIVELOG;
SQL> ALTER DATABASE OPEN;
```

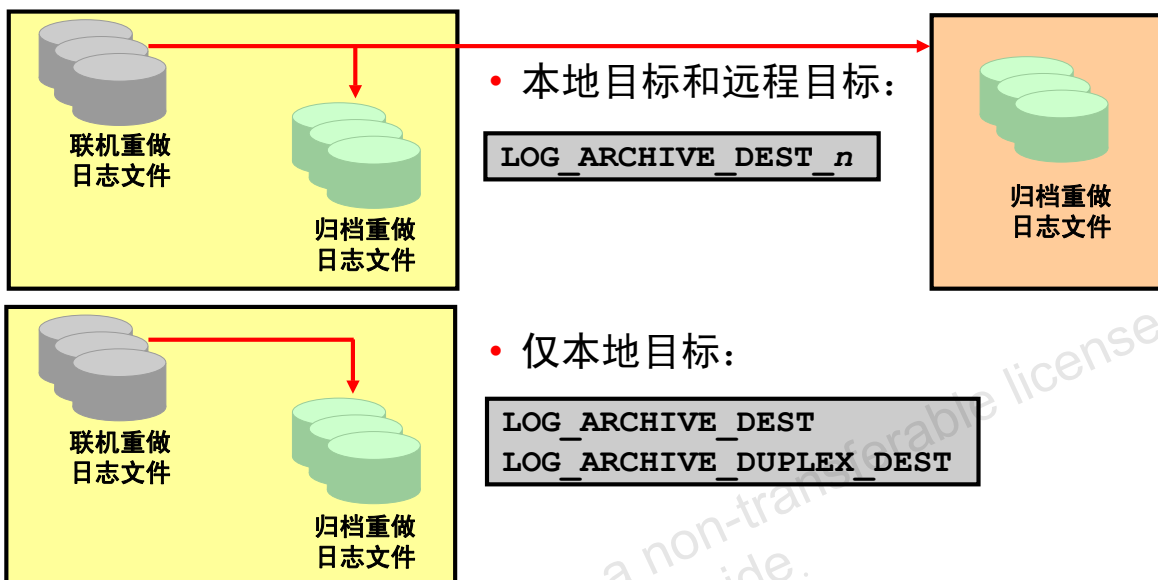
数据库处于 NOARCHIVELOG 模式（默认模式）时，只能恢复到最后一次备份时的状态。在该备份之后执行的所有事务处理都会丢失。

在 ARCHIVELOG 模式下，可一直恢复到最后一次提交时的状态。多数生产数据库都在 ARCHIVELOG 模式下运行。

注：请在切换到 ARCHIVELOG 模式后备份数据库，原因是数据库只能从在该模式下执行的第一次备份进行恢复。

配置归档日志目标位置

最佳做法提示：创建多个目标。如果只有一个目标且将填满，则数据库将停止。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置归档日志目标位置

最佳做法提示：应该创建多个归档日志目标位置，因为如果只有一个目标且将填满，则数据库将停止。

本地目标和远程目标：通过设置 `LOG_ARCHIVE_DEST_n` 初始化参数的集合指定本地目标和远程目标。共有十个初始化参数，因此 n 可以是 1 到 10。

- 为了指定本地存储位置，应为其中一个变量的值提供本地目录名称，即提供“LOCATION=”字符串。例如，要指定 `/disk3/arch` 目录，请按以下方式设置其中一个变量：

```
LOG_ARCHIVE_DEST_1 = 'LOCATION=/disk3/arch'
```

- 如果要为备用数据库指定远程位置，请在值中使用 `SERVICE` 关键字，如下例所示，其中 `standby1` 是备用数据库实例的网络服务名：

```
LOG_ARCHIVE_DEST_2 = 'SERVICE=standby1'.
```

仅本地目标：使用标准版的 Oracle DB，可以将 `LOG_ARCHIVE_DEST` 和 `LOG_ARCHIVE_DUPLEX_DEST` 参数设置为本地磁盘目录。因此，最多可以有两个归档日志文件位置。例如：

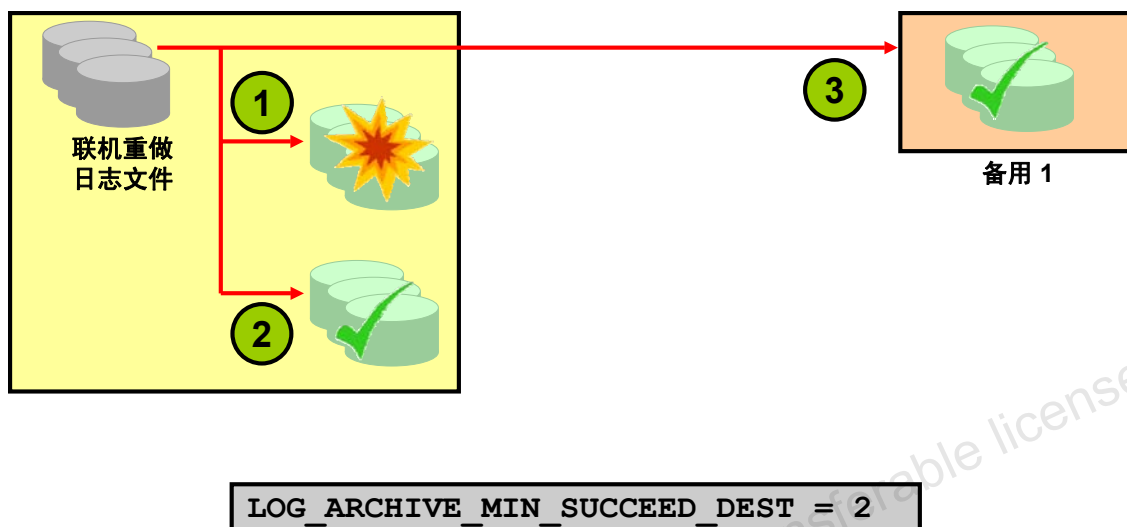
```
LOG_ARCHIVE_DEST = '/disk1/arch'
```

```
LOG_ARCHIVE_DUPLEX_DEST = '/disk2/arch'
```


配置归档日志目标（续）

如果适用，Oracle 建议使用 LOG_ARCHIVE_DEST_ *n* 方法，因为此方法在目标类型和目标数量方面具有最大的灵活度。

确保归档日志成功



版权所有 © 2011, Oracle。保留所有权利。

确保归档日志成功

如果为归档日志文件指定了多个目标，则应指定至少有多少个目标获得成功才认为归档是成功的。可使用 LOG_ARCHIVE_MIN_SUCCEED_DEST 初始化参数执行此操作。将其设置为必须成功接收归档日志文件的目标数量。只有满足此数量，才可以重用联机日志文件。

在幻灯片的示例中，有三个指定的目标：两个是本地目标，一个是远程目标。

LOG_ARCHIVE_MIN_SUCCEED_DEST 设置为 2，表示只要至少有两个目标成功，即可覆盖联机重做日志文件。该示例显示目标 1 已失败。这并不会使数据库停止，因为其中两个已成功。

可以在上一幻灯片所述的任一模型中使用此参数。如果在 LOG_ARCHIVE_DEST_n 模型中使用此参数，则此参数的值可以介于 1 到 10 之间。如果在 LOG_ARCHIVE_DEST 模型中使用此参数，则值可以为 1 或 2，因为在这种情况下只能指定两个目标。

确保归档日志成功（续）

指定 **MANDATORY** 和 **OPTIONAL**

定义某个目标时，可以指定该目标为必需目标。在指定位置后，通过指定 **MANDATORY** 或 **OPTIONAL** 关键字可实现上述目的。下面是一个示例：

```
LOG_ARCHIVE_DEST_1 = 'LOCATION=/disk3/arch MANDATORY'
```

默认值为 **OPTIONAL**。

系统尤为关注必需目标。如果任一必需目标失败，则 Oracle DB 认为日志归档尚未成功，不允许覆盖联机重做日志文件。在此情况下，它将忽略

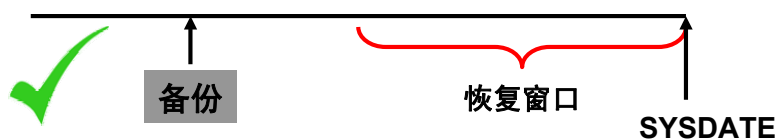
LOG_ARCHIVE_MIN_SUCCEED_DEST 参数。

LOG_ARCHIVE_DEST 指定的任何目标都是必需的。如果

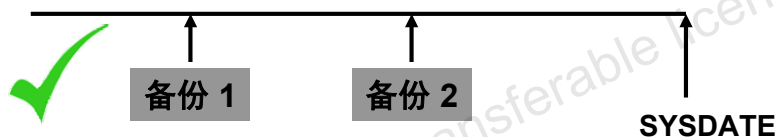
LOG_ARCHIVE_MIN_SUCCEED_DEST = 1，则 **LOG_ARCHIVE_DUPLEX_DEST** 声明的所有目标都是可选的，如果 **LOG_ARCHIVE_MIN_SUCCEED_DEST** = 2，则目标都是必需的。

指定保留策略

- 保留策略：说明要保留的备份及保留的时间长度。
- 有两类保留策略：
 - 恢复窗口：建立一个时间段，在该时段内必须可以进行时间点恢复



- 冗余：确定必须保留的备份的数量（固定值）



- 这两类保留策略互相排斥。

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

指定保留策略

“保留策略”说明了要保留的备份及保留的时间长度。可以通过使用 RMAN 的 CONFIGURE 命令或 Oracle Enterprise Manager 来设置保留策略的值。

恢复窗口保留策略

最佳做法是确定一个时段，在此期间内可以发现逻辑错误，然后通过执行时间点恢复正好恢复到错误前的那一点来修复受影响的对象。此时间段称为恢复窗口。此策略用天数指定。对于每个数据文件，都必须始终存在至少一个满足以下条件的备份：

$$\text{SYSDATE} - \text{backup_checkpoint_time} \geq \text{recovery_window}$$

可以使用下列命令语法配置恢复窗口保留策略：

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF <天数>
DAYS;
```

其中 <天数> 是恢复窗口的长度。

指定保留策略（续）

如果未使用恢复目录，则要防止控制文件中旧的备份记录被覆盖，恢复窗口时段应小于等于控制文件参数 `CONTROL_FILE_RECORD_KEEP_TIME` 的值。如果正在使用恢复目录，则应确保 `CONTROL_FILE_RECORD_KEEP_TIME` 大于目录重新同步的间隔。重新同步在执行以下任务时发生：

- 创建备份时。在此情况下，同步是隐式发生的。
- 执行 `RESYNC CATALOG` 命令。

“使用 RMAN 恢复目录”一课会详细介绍恢复目录。

冗余保留策略

如果需要保留一定数量的备份，可以通过冗余选项设置保留策略。此选项要求在任何备份被标识为过时之前将指定数量的备份列入目录。默认保留策略的冗余度为 1，这表示在任意指定时间一个文件只存在一个备份。当同一文件的最新版本已经有备份时，上一个备份就被视为过时。

可以使用下列命令重新配置冗余保留策略：

```
RMAN> CONFIGURE RETENTION POLICY TO REDUNDANCY <副本数>;
```

其中 <副本数> 是满足策略所需的副本数量。

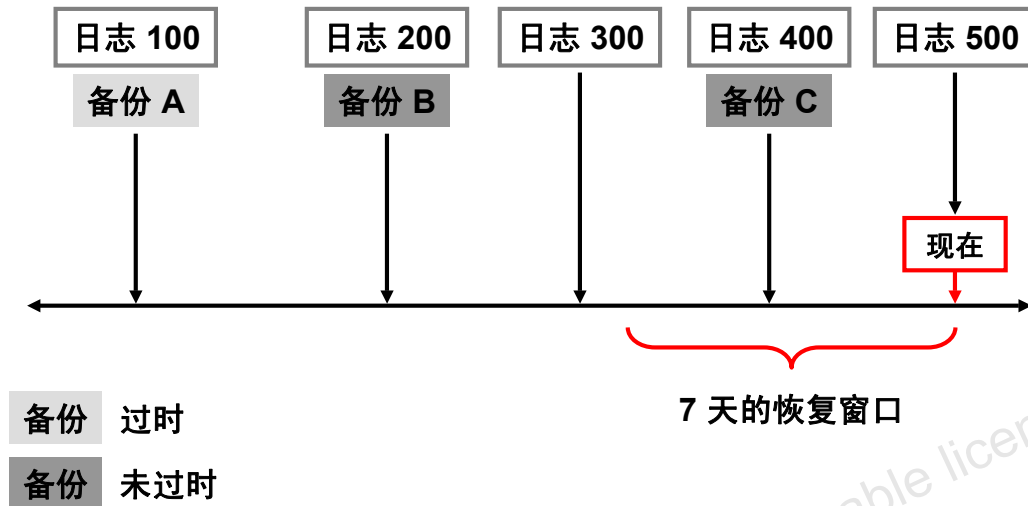
禁用保留策略

您可能希望完全禁用保留策略。如果还有 RMAN 以外的其它系统，并且磁盘备份被备份到磁带中，则可能需要这样做。如果禁用了保留策略，则 RMAN 不会认为某个备份过时。因为 RMAN 不必决定何时从磁盘中删除备份（因为有其它实用程序管理此任务），所以不需要对 RMAN 进行此方面的配置。在此情况下，维护每个备份的记录的时间长度即为 `CONTROL_FILE_RECORD_KEEP_TIME` 初始化参数指定的时间长度。请使用下列命令禁用保留策略：

```
RMAN> CONFIGURE RETENTION POLICY TO NONE;
```

注：可以指定某个备份不遵从所定义的保留策略。这称为归档备份，“使用 RMAN 创建备份”一课会对此进行介绍。

恢复窗口保留策略：示例



需要备份 B 和归档日志 201 到 500 才能满足此保留策略。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

指定恢复窗口保留策略：示例

本幻灯片中的保留策略显示了需要恢复到过去七天内的任何时间的情况。某些备份和日志已过时，因为恢复到七天窗口内的某一时间不需要使用这些备份和日志。此保留策略的配置如下：

```
RMAN> CONFIGURE RETENTION POLICY TO RECOVERY WINDOW OF 7 DAYS;
```

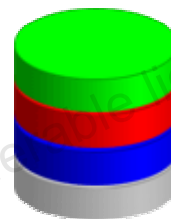
如果备份和归档日志文件都可用，则恢复到恢复窗口内某个时间点所需的数据仅为备份 B 以及日志 201 到 500。注意，不需要备份 A，因为有一个更新的备份 (B) 仍在恢复窗口之前。另外，不能只保留备份 C，因为它无法满足恢复到恢复窗口开始时的时间点的需要。所需的数据是在恢复窗口开始之前创建的最后一个备份以及该备份之后的所有日志。

使用快速恢复区

- 永久项目：
 - 当前控制文件的多路复用副本
 - 联机重做日志的多路复用副本
- 临时项目：
 - 归档重做日志
 - 数据文件副本
 - 控制文件副本
 - 控制文件自动备份
 - 备份片段
 - 闪回日志



数据库



快速恢复区

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用快速恢复区

快速恢复区是 Oracle DB 中所有与恢复相关的文件和活动的统一存储位置。在发生介质故障后完全恢复数据库所需的所有文件都包含在快速恢复区中。与恢复相关的文件有两种类型：永久文件和临时文件。永久文件由实例主动使用。临时文件只有在执行某类恢复操作时才需要。

永久项目

- **控制文件：**根据几个初始化参数的设置，控制文件的副本是在创建新数据库或控制文件时在快速恢复区中创建的。有关详细资料，请参阅《Oracle Database SQL Language Reference》中 CREATE CONTROLFILE 命令的“Semantics”一节。
- **联机重做日志文件的多路复用副本：**可包含每个重做日志组的镜像副本。创建数据库时，可以使用 LOGFILE 子句指定联机重做日志文件的位置。如果未包含该子句，则系统会根据下列初始化参数的值设置位置：
 - **DB_CREATE_ONLINE_LOG_DEST_n：**如果设置了这些变量中的一个或多个，则只使用这些位置。
 - **DB_CREATE_FILE_DEST：**如果设置了此参数，则该位置是主要文件位置。
 - **DB_RECOVERY_FILE_DEST：**如果除 DB_CREATE_FILE_DEST 外还设置了此参数，则此位置将用作镜像。

使用快速恢复区（续）

有关这些变量如何影响联机重做日志的位置的详细资料，请参阅《Oracle Database SQL Language Reference》中 CREATE DATABASE 语句的 LOGFILE 子句。

临时项目

- **归档重做日志文件：**配置了快速恢复区后，LOG_ARCHIVE_DEST_1 会被自动设置为快速恢复区位置。归档进程后台进程会在快速恢复区和所配置的其它 LOG_ARCHIVE_DEST_n 位置中创建归档重做日志文件。如果未定义 LOG_ARCHIVE_DEST_n 位置，则归档重做日志文件的默认位置是快速恢复区。
- **闪回日志：**启用闪回数据库时会生成闪回日志。
- **控制文件自动备份：**由 RMAN 创建的控制文件自动备份和由 Oracle DB Server 生成的自动备份的默认位置是快速恢复区。
- **数据文件副本：**BACKUP AS COPY 命令在快速恢复区中创建映像数据文件副本。
- **RMAN 文件：**快速恢复区是 RMAN 用于备份和通过恢复操作从磁带还原归档日志内容的默认位置。

注：如果您的 FRA 需要高性能，请考虑将其创建在其自己的物理磁盘和控制器上。

定义快速恢复区

FRA 是通过设置以下两个动态初始化参数来定义的：

- **DB_RECOVERY_FILE_DEST_SIZE**：设置磁盘限制
- **DB_RECOVERY_FILE_DEST**：设置 FRA 的位置

The screenshot displays the Oracle Enterprise Manager interface for configuring the Flash Recovery Area (FRA). It includes the ARCHIVELOG Mode settings, a table for Archived Redo Log Destinations, and a detailed view of the FRA location and size. A pie chart illustrates the usage of the FRA space.

Number	Archived Redo Log Destination	Status	Type
1	USE_DB_RECOVERY_FILE_DEST	VALID	Local

Flash Recovery Area Usage

Component	Size (GB)	Percentage
Usable	4.11	68.5%
Backup Piece	1.34	22.4%
Archived Redo Log	0.4	6.6%
Online Log	0.15	2.5%
Control File	0	0%
Image Copy	0	0%
Flashback Log	0	0%

Flash Recovery Area Location: +FRA
Flash Recovery Area Size: 6 GB
 Flash Recovery Area Size must be set when the location is set.

Flash Recovery Area Summary:
 Non-reclaimable Flash Recovery Area (GB): 1.89
 Reclaimable Flash Recovery Area (MB): 609
 Free Flash Recovery Area (GB): 3.51

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

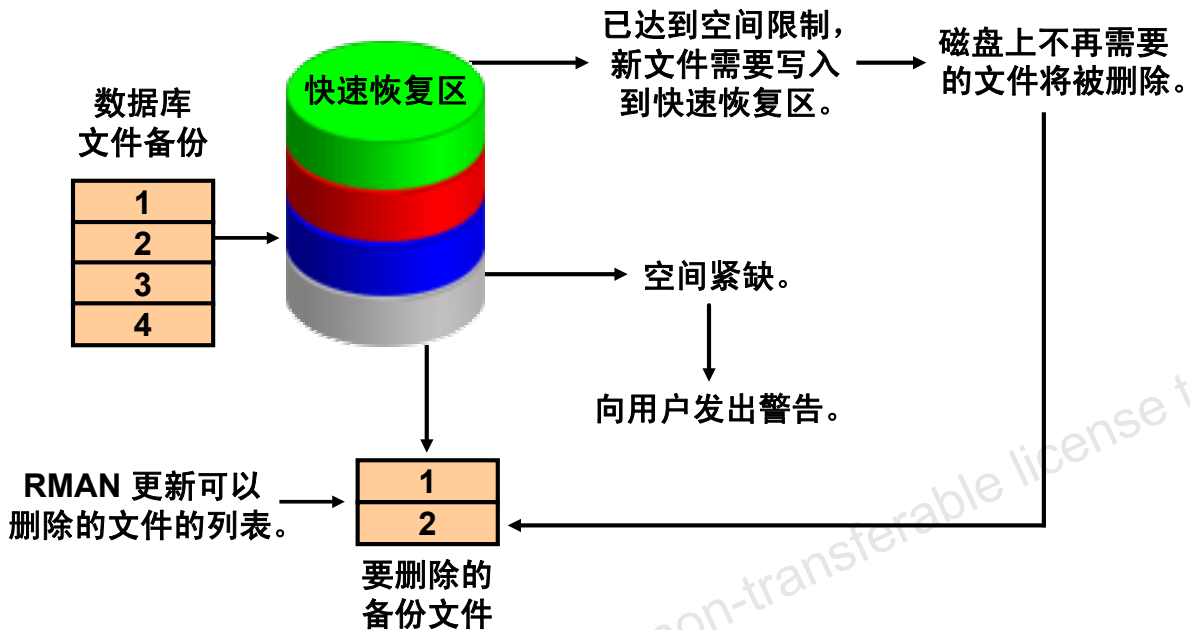
定义快速恢复区

请使用以下必需参数定义 FRA：

- **DB_RECOVERY_FILE_DEST_SIZE**：必须定义磁盘限制，这是允许 FRA 使用的空间量。通过设置限制可留出一部分磁盘空间作为它用，而不被 FRA 所专用。
 - 对于磁盘限制大小，基本的建议是使其等于数据库大小、增量备份大小和尚未复制到磁带的所有归档日志文件的大小的总和。给出该建议是因为 Oracle 建议的备份策略先执行数据库的一个映像副本（不包含临时文件），然后执行增量备份。
 - FRA 的最小大小应至少足够放得下尚未复制到磁带的归档重做日志文件。
 - FRA 的大小取决于备份策略和所实施的其它选项。保证还原点也会影响 FRA 的大小。
- **DB_RECOVERY_FILE_DEST**：FRA 说明中包含一个位置，这个位置是用于创建文件的有效目标。

可以使用 Oracle Enterprise Manager Grid Control 和 Database Control 轻松地定义 FRA。导航到“Availability > Recovery Settings（可用性 > 恢复设置）”。可以在“Recovery Settings（恢复设置）”页上定义 FRA 的位置和大小。在指定 FRA 的位置时必须设置其大小。

快速恢复区空间管理



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

快速恢复区空间管理

RMAN 每次在快速恢复区中创建文件时，都会更新磁盘上不再需要的文件的列表。根据 `DB_RECOVERY_FILE_DEST_SIZE` 的值，当快速恢复区因没有可删除的文件而出现空间紧缺或空闲空间不足的情况时，系统会警告您存在磁盘空间不足的危险。这时，Oracle DB Server 和 RMAN 会继续在快速恢复区中创建文件，直到达到 100% 磁盘限制。设置 `DB_RECOVERY_FILE_DEST_SIZE` 时，必须分配足够的空间来存放恢复文件，包括等待备份到磁带的备份。为了提供空闲空间，过时或已备份到磁带的文件都可能被删除。将某一文件写入到快速恢复区时，如果需要为该文件腾出空间，Oracle DB Server 会删除过时文件列表中的某个文件。在快速恢复区中写入和删除文件时，会向预警日志写入通知。

注：当快速恢复区的已用空间达到 85% 时，会发出警告；当已用空间达到 97% 时，会发出严重警告。这些都是内部设置，无法更改。

以下是一个预警日志输出示例：

```
WARNING: db_recovery_file_dest_size of 52428800 bytes is
100.00% used, and has 0 remaining bytes available.
```

快速恢复区空间管理（续）

可以发出以下查询来确定要执行的操作：

```
SQL> SELECT object_type, message_type, message_level,  
2 reason, suggested_action  
3 FROM dba_outstanding_alerts;
```

可以选择增加额外的磁盘空间、将文件备份到三级存储设备、使用 RMAN 从快速恢复区删除文件或考虑更改 RMAN 保留策略。

快速恢复区空间使用情况

- 将保留策略配置为适合数据库的最小值。
- 定期备份归档日志文件并在完成备份后删除这些文件。
- 此外，还可以配置归档重做日志删除策略。
- 使用 RMAN REPORT OBSOLETE 和 DELETE OBSOLETE 命令删除不需要的备份和文件副本。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

快速恢复区空间使用情况

为避免出现快速恢复区空间不足的情况，请根据需要或在适用情况下执行以下步骤：

- 使用 RMAN 从快速恢复区删除不需要的文件。
- 使用 RMAN 经常备份快速恢复区。
- 更改 RMAN 保留策略，缩短备份的保留期。
- 更改 RMAN 归档日志删除策略。
- 如果经常出现空间不足的情况，请增加磁盘空间并增大 DB_RECOVERY_FILE_DEST_SIZE 数据库初始化参数值。

Oracle Enterprise Manager 不报告快速恢复区在磁盘上使用的空间量，也不报告在快速恢复区目录树中使用的空间量，但报告 RMAN 认为位于目录中的文件的大小。所以不要将任何文件放在此不受 RMAN 管理的区域。

如果使用 RMAN 之外的工具从此区域删除任何文件，请使用 RMAN 从目录中删除文件项。例如，要备份快速恢复区中的归档日志文件，然后在成功备份这些文件后将其删除，应使用如下 RMAN 命令：

```
BACKUP ARCHIVELOG ALL DELETE ALL INPUT;
```

快速恢复区空间使用情况（续）

即使使用 RMAN 之外的备份解决方案，仍需要使用 RMAN 从快速恢复区删除这些文件。备份归档重做日志文件并将这些文件从磁盘中删除后，应使用 RMAN 的 CROSSCHECK 和 DELETE 命令从快速恢复区回收归档日志空间。这个操作应定期执行或在每次备份后执行。还可以使用 Oracle Enterprise Manager 的“Manage Backups（管理备份）”页来管理备份。可以在该页上执行交叉检查操作，还可以删除失效和过时的备份。

配置归档重做日志删除策略

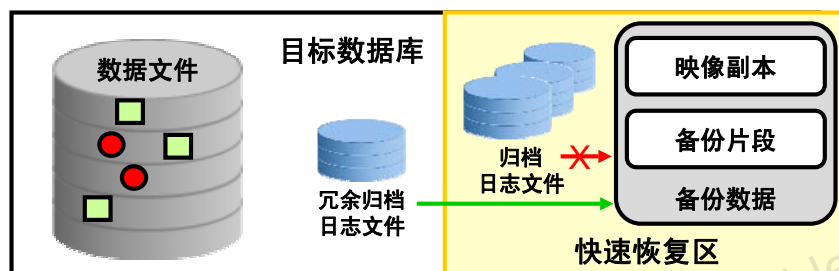
可以使用 CONFIGURE ARCHIVELOG DELETION POLICY 命令指定归档重做日志何时才能删除。该删除策略适用于所有归档目标，包括 FRA。

归档重做日志可由数据库自动删除，或因用户启动的 RMAN 命令而删除：

- 只有 FRA 中的日志可由数据库自动删除。对于 FRA 中的归档重做日志文件，数据库将尽可能长地保留这些文件，并在需要额外磁盘空间时自动删除符合条件的日志。
- 发出 BACKUP ... DELETE INPUT 或 DELETE ARCHIVELOG 时，可以从任何位置（无论在 FRA 内部还是外部）手动删除符合条件的日志。
- 默认值为：
CONFIGURE ARCHIVELOG DELETION POLICY TO NONE;

自动为您完成的任务

- 简化了多组件环境中的归档日志管理
- 通过将备份故障转移到可选目标提高了可用性



ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

自动为您完成的任务

简化了多组件环境中的归档日志管理

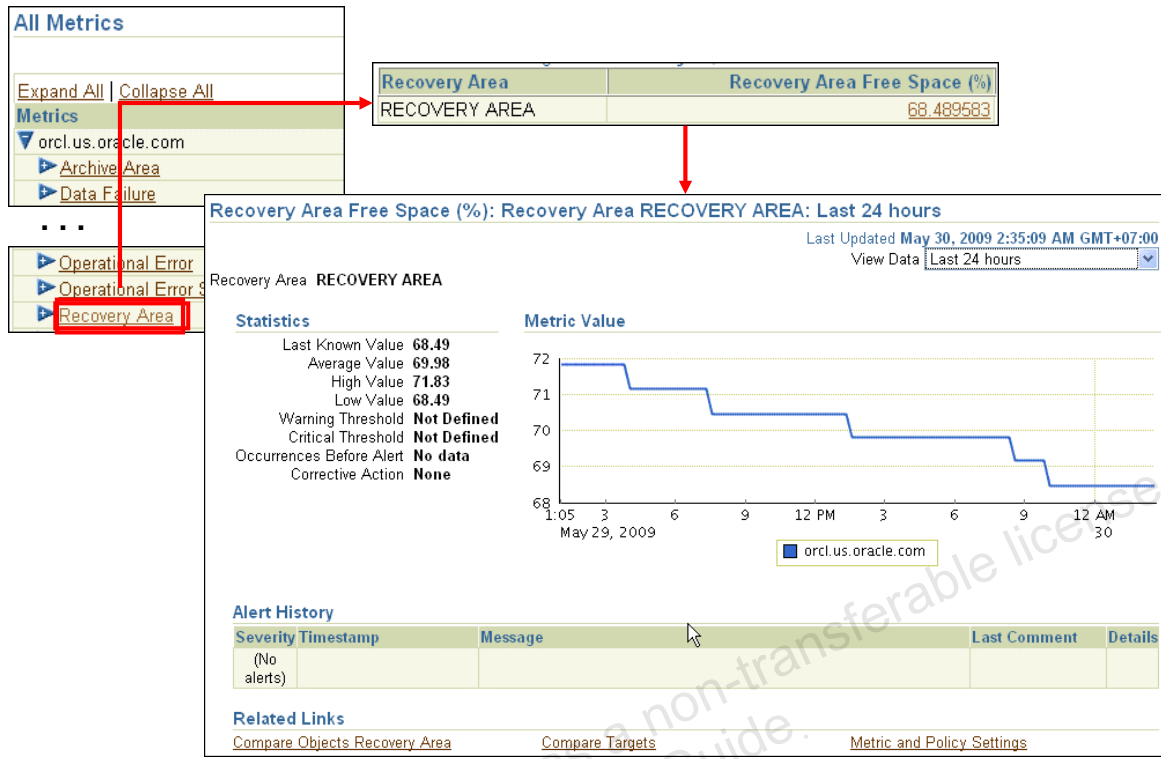
当归档日志由多个组件（例如 Streams）使用时，此功能简化了归档日志管理。它还提高了当快速恢复区中的归档日志丢失或无法访问时，备份归档日志时的可用性。

增强了删除策略的配置

仅当任何必需的组件（如 Data Guard、Streams、Flashback Database，等等）都不需要归档重做日志时，才能将归档重做日志删除。配置了归档日志删除策略后，该配置会应用到所有归档目标，包括 FRA。BACKUP ... DELETE INPUT 和 DELETE ... ARCHIVELOG 都使用此配置，FRA 也是。

备份恢复区时，如果 FRA 中的归档重做日志无法访问或已损坏，则 RMAN 可故障转移到其它归档重做日志目标位置。

监视 FRA



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

监视 FRA

通过 Oracle Enterprise Manager Database Control 可以查看实时 FRA 度量值。在主页上，向下滚动到“Related Links（相关链接）”部分，选择“All Metrics（所有度量）”。浏览列表，然后单击“Recovery Area（恢复区）”。

此时出现的页面将显示“Recovery Area Free Space (%)（恢复区空闲空间 (%)）”度量，该度量指示恢复区空闲空间的百分比。单击该百分比数字可查看恢复区使用情况。

使用快速恢复区的优点

为与恢复相关的文件使用快速恢复区具有以下优点：

- 便于确定数据库备份的位置
- 自动管理为恢复文件分配的磁盘空间

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用快速恢复区的优点

为与恢复相关的所有文件使用 FRA 可简化数据库的日常管理工作。

Oracle Corporation 建议为所有与恢复相关的文件使用快速恢复区。

测验

可以为以下哪项创建备份集？

1. 数据文件
2. 归档日志
3. 联机日志
4. 备份集

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1、2、4

测验

选择关于快速恢复区 (FRA) 的所有正确语句：

1. FRA 可以使用 ASM 磁盘组。
2. FRA 可以使用操作系统目录。
3. FRA 仅可以由一个数据库使用。
4. 应该将 FRA 放置在速度最慢的磁盘上以提高恢复能力。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1、2

小结

在本课中，您应该已经学会：

- 调用和配置 RMAN
- 在 ARCHIVELOG 模式下配置数据库
- 配置多个归档日志文件目标以提高可用性
- 配置快速恢复区
- 指定保留策略
- 描述快速恢复区的优点

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 2 概览：进行配置以实现可恢复性

本练习包含以下主题：

- 将数据库置于 ARCHIVELOG 模式
- 验证是否配置了 FRA
- 使用 RMAN 连接到目标数据库

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

3 使用 RMAN 恢复目录

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

学完本课后，应能完成以下工作：

- 对恢复目录和 RMAN 资料档案库控制文件的使用进行比较
- 创建和配置恢复目录
- 在恢复目录中注册数据库
- 同步恢复目录
- 使用 RMAN 存储脚本
- 备份恢复目录
- 创建虚拟专用目录

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

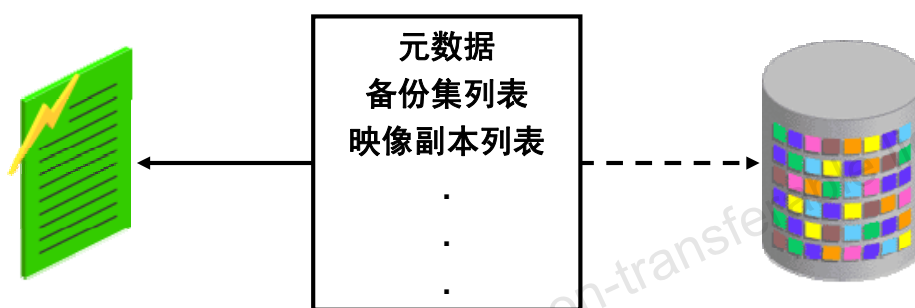
RMAN 资料档案库数据存储：比较选项

控制文件：

- 管理更简单
- 默认设置

恢复目录：

- 复制控制文件数据
- 存储更长的备份历史记录
- 服务于多个目标
- 存储 RMAN 脚本



版权所有 © 2011, Oracle。保留所有权利。

ORACLE

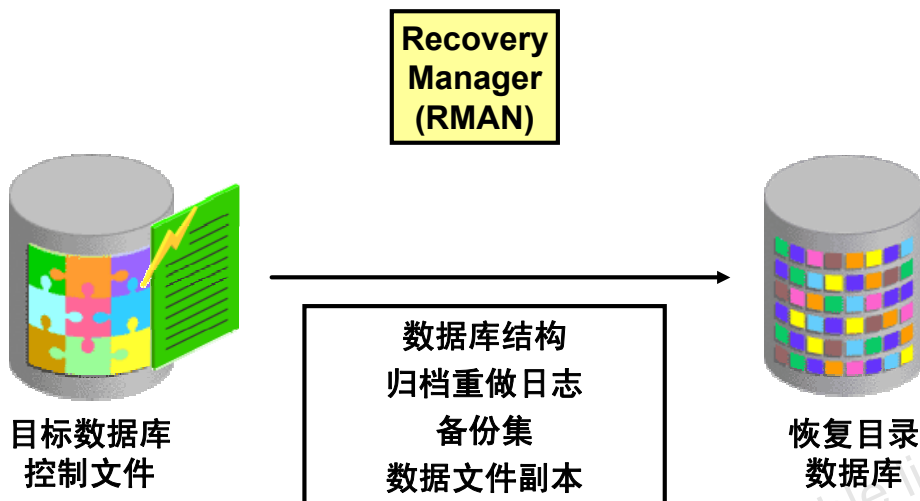
RMAN 资料档案库数据存储：比较选项

RMAN 资料档案库数据始终存储在目标数据库的控制文件中。此外，它也可以存储在一个名为“恢复目录”的单独数据库中。

恢复目录会在单独的数据库中保留备份信息，这在控制文件丢失时尤为有用。这样您可以存储更长的备份历史记录，比基于控制文件的资料档案库所存储的历史记录还要长。使用一个恢复目录可存储多个目标数据库的信息。恢复目录还可用来存放 RMAN 存储脚本，这些脚本是 RMAN 命令的序列。

如果您的备份管理要求很简单，则 Oracle 建议您使用控制文件选项而不要使用恢复目录。拥有恢复目录意味着您需要管理并备份其它数据库。因此，只有在可利用恢复目录所提供的优点（如较长的备份保留时间）时，才使用恢复目录。

在恢复目录中存储信息



版权所有 © 2011, Oracle。保留所有权利。

在恢复目录中存储信息

在执行更新资料档案库的任何操作之后以及执行某些操作之前，RMAN 会将关于数据库结构、归档重做日志、备份集和数据文件副本的信息从目标数据库控制文件中传播到恢复目录中。

使用恢复目录的原因

- 与控制文件相比可以存储更多的历史信息
- 可使用 RMAN 存储脚本
- 可为所有已注册目标创建自定义报告
- 可以使用 BACKUP 命令的 KEEP FOREVER 子句
- 可以列出在给定时间位于或曾经位于目标数据库中的数据文件和表空间

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用恢复目录的原因

虽然可以使用控制文件作为 RMAN 的唯一资料档案库，但是控制文件用于存储备份活动记录的空间有限。使用恢复目录时，可以存储更长的备份历史记录。这样可以恢复到比控制文件中的历史记录更早的时间。

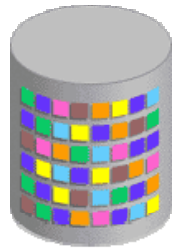
如果要使用 RMAN 存储脚本，就必须使用恢复目录。

使用恢复目录时，所有已注册目标的备份和恢复信息都存放在一个位置，因此可通过以恢复目录所有者的身份连接并查询各个 RC_ 视图来创建自定义报告。如果不使用恢复目录，就必须单独连接到每个目标数据库实例并查询 V\$ 视图，才能了解目标控制文件中的 RMAN 信息。

可以使用 BACKUP ... KEEP 命令创建备份并使其保留时长不同于所配置的保留策略指定的时长。KEEP FOREVER 子句指定备份或副本永不失效并要求使用恢复目录，以便备份记录可无限期保留。

REPORT SCHEMA 命令列出目标数据库中的表空间和数据文件。如果添加 AT [time|scn|logseq] 选项，则可以查看过去某个时间的信息。只有在使用恢复目录时才能使用 AT 选项。

创建恢复目录：三个步骤



配置恢复目录
数据库。



创建恢复目录
所有者。



创建恢复目录。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

创建恢复目录：三个步骤

要创建恢复目录，请执行以下三个步骤：

1. 配置要在其中存储恢复目录的数据库。
2. 创建恢复目录所有者。
3. 创建恢复目录。

配置恢复目录数据库

- 分配恢复目录使用的空间。以下是需要考虑的因素：
 - 恢复目录支持的数据库数
 - 记录的归档重做日志文件数和备份数
 - 使用 RMAN 存储脚本
- 创建恢复目录所用的表空间，这将成为恢复目录所有者的默认表空间。

```
SQL> CREATE TABLESPACE rcat_ts DATAFILE SIZE 15M;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置恢复目录数据库

确定要在其中安装恢复目录方案的数据库。请务必考虑此数据库的备份和恢复过程。本课程稍后将对备份恢复目录进行介绍。

恢复目录方案所需的空間量取决于恢复目录监视的数据库数。所需空间随每个数据库的归档重做日志文件数和备份数的增加而增加。如果使用 RMAN 存储脚本，则必须分配这些脚本使用的空间。在恢复目录中注册的每个数据库的示例空间需求为 15 MB。

创建恢复目录所有者



- 创建恢复目录所有者。
- 授予 RECOVERY_CATALOG_OWNER 角色。



```
SQL> CREATE USER rcowner IDENTIFIED BY rcpass
2 TEMPORARY TABLESPACE temp
3 DEFAULT TABLESPACE rcat_ts
4 QUOTA UNLIMITED ON rcat_ts;
SQL> GRANT recovery_catalog_owner TO rcowner;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

创建恢复目录所有者

创建一个作为恢复目录所有者的用户。将此用户的默认表空间设置为您为恢复目录创建的表空间。请务必为该用户提供对该表空间的 UNLIMITED 限额。创建用户之后，请为其授予 RECOVERY_CATALOG_OWNER 角色。RECOVERY_CATALOG_OWNER 角色提供恢复目录所有者的权限。该角色包含下列系统权限：ALTER SESSION、CREATE CLUSTER、CREATE DATABASE LINK、CREATE PROCEDURE、CREATE SEQUENCE、CREATE SESSION、CREATE SYNONYM、CREATE TABLE、CREATE TRIGGER、CREATE TYPE 和 CREATE VIEW。

可以使用 SQL 命令或 Oracle Enterprise Manager 创建用户并授予角色。

创建恢复目录

- 以目录所有者的身份连接到恢复目录数据库:

```
$ rman  
RMAN> CONNECT CATALOG username/password@net_service_name
```

- 执行 CREATE CATALOG 命令:

```
RMAN> CREATE CATALOG;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

创建恢复目录

创建目录所有者后, 请使用 RMAN CREATE CATALOG 命令在目录所有者的默认表空间中创建目录表。

注: 与任何数据库一样, 如果 ORACLE_SID 环境变量设置为恢复目录数据库的 SID, 则不需要在 CONNECT 语句中提供 net_service_name。

管理恢复目录中的目标数据库记录

- 在恢复目录中注册目标数据库
- 将其它备份文件列入目录
- 从恢复目录中注销目标数据库

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理恢复目录中的目标数据库记录

虽然大多数信息会自动从控制文件传播到恢复目录中，但还是需要执行一些操作才能在恢复目录中维护目标数据库记录。

在恢复目录中注册数据库

RMAN 会执行下列操作：

- 在恢复目录表中创建目标数据库的行
- 将目标数据库控制文件中的数据复制到恢复目录表
- 使恢复目录与控制文件同步

```
$ rman TARGET / CATALOG  
    username/password@net_service_name  
RMAN> REGISTER DATABASE;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

在恢复目录中注册数据库

创建恢复目录后，必须在恢复目录中注册目标数据库。要注册目标数据库，请执行以下步骤：

1. 调用 RMAN 并连接到恢复目录数据库和目标数据库，如下例所示：
 % rman TARGET / CATALOG rman/rman@reccatdb
2. 确保目标数据库已装载或已打开。
3. 发出 REGISTER 命令，在恢复目录中注册目标数据库：
 RMAN> REGISTER DATABASE;

使用 Oracle Enterprise Manager 注册数据库

要在恢复目录中注册数据库，请使用 Oracle Enterprise Manager (EM) 执行下列步骤：

1. 针对目标数据库运行 EM，并导航到“Recovery Catalog Settings（恢复目录设置）”页。
2. 如果该页上没有列出恢复目录，请将其添加到配置中。
3. 指定目标数据库使用从列表中选择恢复目录。



EM 的注册方法还会导致 EM 对与备份和恢复相关的操作使用恢复目录。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 Oracle Enterprise Manager 注册数据库

要使用 EM 在恢复目录中注册数据库，必须先将恢复目录添加到 EM 配置中。在目标数据库上运行 EM，然后选择该恢复目录作为目标数据库的恢复目录。

如果使用 RMAN 注册数据库，而且没有执行幻灯片中的步骤，则使用 EM 执行的任何备份和恢复操作将都不使用恢复目录。因此，如果计划使用 EM，请执行此处介绍的注册步骤，即使已执行了 RMAN REGISTER DATABASE 命令也是如此。

在 Oracle Enterprise Manager 中：

1. 在“EM Database（EM 数据库）”主页中，导航到“Availability > Recovery Catalog Settings（可用性 > 恢复目录设置）”。单击“Add Recovery Catalog（添加恢复目录）”，指定数据库的主机、端口和 SID 及现有恢复目录。
2. 定义了恢复目录数据库后，在“Recovery Catalog Setting（恢复目录设置）”页上选择“Use Recovery Catalog（使用恢复目录）”，可在恢复目录数据库中注册数据库。单击“OK（确定）”，即可在目录中注册数据库。

从恢复目录中注销目标数据库

- 执行此操作会从恢复目录中删除目标数据库的相关信息。
- 当您不希望目标数据库的定义继续存在于恢复目录中时，请执行此操作。

```
$ rman TARGET / CATALOG  
    username/password@net_service_name  
RMAN> UNREGISTER DATABASE;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

从恢复目录中注销目标数据库

从恢复目录中注销某一数据库时，恢复目录中的所有 RMAN 资料档案库记录都会丢失。您可以重新注册这个数据库。此时，该数据库的恢复目录记录取决于再次注册时控制文件的内容。

通常，只有不再需要使用数据库的恢复目录时或数据库不再存在时，才会注销目标数据库。

注：如果是使用 Oracle Enterprise Manager Database Control 注册的数据库，则也必须使用它来注销数据库。

将其它备份文件列入目录

- CATALOG 可用于将不再在控制文件中列出的现有备份文件列入目录。
- 这样，RMAN 即可在执行还原操作期间使用这些文件。
- 使用 CATALOG 命令可将以下类型的备份文件添加到恢复目录中：
 - 控制文件副本
 - 数据文件副本
 - 备份片段
 - 归档重做日志文件

```
RMAN> CATALOG BACKUPPIECE 'file_name';
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

将其它备份文件列入目录

如果磁盘上有其它控制文件副本、数据文件副本、备份片段或归档重做日志文件，则可使用 CATALOG 命令将其列入恢复目录。如果某些备份因时间太早已从控制文件中清除，则可将这些备份列入目录，以便 RMAN 在执行还原操作期间使用它们。

以下是将控制文件、数据文件、归档重做日志文件和备份片段列入目录的示例：

```
RMAN> CATALOG CONTROLFILECOPY
'/disk1/controlfile_bkup/2009_01_01/control01.ctl';
RMAN> CATALOG DATAFILECOPY
'/disk1/datafile_bkup/2009_01_01/users01.dbf';
RMAN> CATALOG ARCHIVELOG '/disk1/arch_logs/archive1_731.log',
'/disk1/arch_logs/archive1_732.log';
RMAN> CATALOG BACKUPPIECE '/disk1/backups/backup_820.bkp';
```

可以将当前启用的快速恢复区中的所有文件列入目录，如下所示：

```
RMAN> CATALOG RECOVERY AREA NOPROMPT;
```

START WITH 选项

使用 START WITH 选项可将指定目录树中找到的所有文件列入目录。提供表示目录的前缀以及用于查找的可能的文件前缀。不能使用通配符，仅提供一个前缀。

将其它备份文件列入目录（续）

指定目录和子目录中找到的所有类型的备份文件都被列入目录。假定在 /tmp/arch_logs 目录中有多个备份文件。下列命令将所有文件列入目录：

```
RMAN> CATALOG START WITH '/tmp/arch_logs/';
```

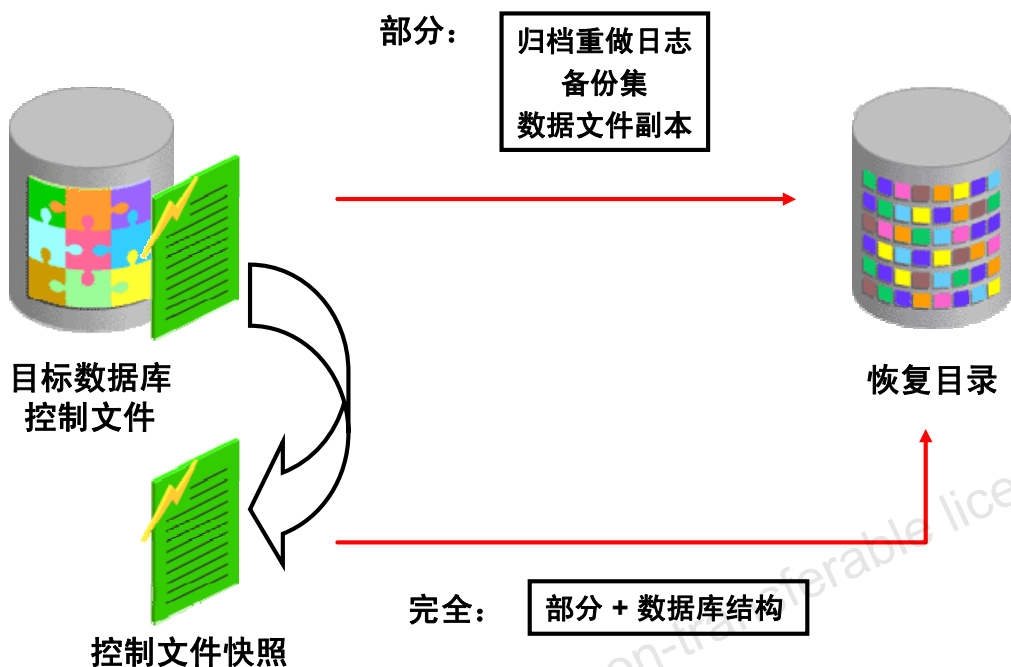
假定您只希望将 /tmp 目录中文件名以字符串 bset 开头的文件列入目录。下列命令可实现此目的：

```
RMAN> CATALOG START WITH '/tmp/bset';
```

此命令还可将在以 /tmp/bset 开头的目录树中找到的所有备份文件列入目录。

CATALOG 命令可以在不连接到恢复目录的情况下使用。

恢复目录重新同步：概念



版权所有 © 2011, Oracle。保留所有权利。

恢复目录重新同步：概念

RMAN 执行重新同步时，会对恢复目录和目标数据库的当前控制文件或备份/备用控制文件进行比较，然后用缺失或更改的信息来更新恢复目录。

重新同步有两种类型：部分和完全。对于部分重新同步，RMAN 会比较控制文件和恢复目录，并使用与备份、归档重做日志、数据文件副本等有关的任何元数据更新恢复目录。对于完全重新同步，RMAN 先创建控制文件快照，该快照只是控制文件的临时副本。它使用快照与恢复目录进行比较。它比较和更新部分重新同步处理的所有数据，还包括所有数据库结构更改。例如，完全重新同步包括数据库方案更改或新表空间。

注：数据库方案包括数据文件、重做日志文件、归档日志文件、还原段的名称和位置以及在控制文件中找到的其它信息。

如果只对控制文件中由 `CONTROL_FILE_RECORD_KEEP_TIME` 控制的记录进行更改，则使用部分重新同步即可。否则，需使用完全重新同步。也可以通过发出 `RESYNC CATALOG` 命令执行完全重新同步，下一张幻灯片将介绍此内容。

手动重新同步恢复目录

可在下列情况下手动重新同步恢复目录：

- RMAN 自动重新同步恢复目录时恢复目录不可用
- 对目标数据库执行不常执行的备份时
- 对目标数据库的物理结构进行更改后

```
RMAN> RESYNC CATALOG;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

手动重新同步恢复目录

可在下列情况下手动重新同步恢复目录：

- 发出 RMAN 命令以执行部分重新同步时恢复目录不可用
- 因为在发生重做日志切换或归档了重做日志时恢复目录未自动进行更新，所以执行不常执行的目标数据库备份
- 对目标数据库的物理结构进行任何更改后

注：有关重新同步期间所更新记录的详细信息，请参阅《Oracle Backup and Recovery User's Guide》。

使用 RMAN 存储脚本

存储脚本是：

- 命令文件的替代方式
- 供可连接到目标数据库和恢复目录的任意 RMAN 客户机使用

- 有两种类型：

- 本地：与创建脚本时 RMAN 连接到的目标数据库相关联

```
CREATE SCRIPT script_name
{ <RMAN commands> }
```

- 全局：可以针对在恢复目录中注册的任何数据库执行

```
CREATE GLOBAL SCRIPT script_name
{ <RMAN commands> }
```

- 从文本文件创建（其它选项）

```
CREATE [GLOBAL] SCRIPT script_name FROM FILE 'file_name';
```

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

使用 RMAN 存储脚本

可以作为命令文件的替代方式使用 RMAN 存储脚本来管理经常使用的 RMAN 命令序列。存储脚本与命令文件不同，命令文件只可在存储它们的系统中使用，存储脚本始终可供连接到目标数据库和恢复目录的任何 RMAN 客户机使用。

存储脚本可以定义为全局存储脚本或本地存储脚本。本地存储脚本与创建脚本时 RMAN 连接到的目标数据库相关联，只有连接到该目标数据库时才能执行本地存储脚本。全局存储脚本则可针对在恢复目录中注册的任何数据库执行，前提是 RMAN 客户机连接到恢复目录和某一目标数据库。

创建 RMAN 存储脚本

连接到所需的数据库和恢复目录，然后执行 CREATE SCRIPT 命令即可创建存储脚本。

执行 RMAN 存储脚本

- 执行脚本：

```
RUN { EXECUTE SCRIPT
      script_name
    ; }
```

- 执行全局脚本：

```
RUN { EXECUTE GLOBAL SCRIPT
      script_name
    ; }
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行 RMAN 存储脚本

连接到目标数据库和恢复目录，然后使用 EXECUTE SCRIPT 命令即可执行存储脚本。请注意，EXECUTE SCRIPT 命令需要 RUN 块。如果脚本中的某一 RMAN 命令失败，则不会执行脚本中的后续 RMAN 命令。

执行脚本时，脚本将使用当时配置的自动通道。如果需要替换已配置的通道，请在脚本中使用 ALLOCATE CHANNEL 命令，如下例所示：

```
RMAN> RUN
{
  ALLOCATE CHANNEL ch1 DEVICE TYPE DISK;
  ALLOCATE CHANNEL ch2 DEVICE TYPE DISK;
  ALLOCATE CHANNEL ch3 DEVICE TYPE DISK;
  EXECUTE SCRIPT full_backup;
}
```

维护 RMAN 存储脚本

- 显示脚本:

```
PRINT [GLOBAL] SCRIPT script_name;
```

- 将脚本内容发送到文件:

```
PRINT [GLOBAL] SCRIPT script_name TO FILE 'file_name';
```

- 显示定义脚本的名称:

```
LIST [GLOBAL] SCRIPT NAMES;
```

- 显示脚本:

```
REPLACE [GLOBAL] SCRIPT script_name  
{ <RMAN commands> ; }
```

- 通过文本文件更新脚本:

```
REPLACE [GLOBAL] SCRIPT script_name FROM FILE  
'file_name';
```

- 删除脚本:

```
DELETE SCRIPT script_name;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

显示 RMAN 存储脚本信息

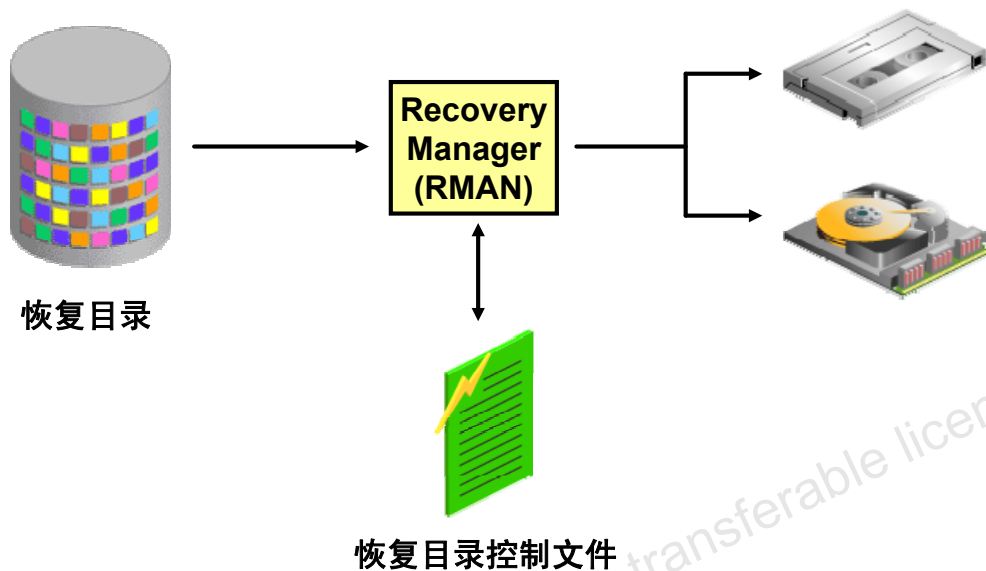
连接到目标数据库和恢复目录，然后使用 PRINT SCRIPT 命令，就可显示存储脚本或将脚本写到文件。

使用 LIST SCRIPT NAMES 命令可显示恢复目录中定义的脚本的名称。这个命令显示可对当前连接的目标数据库执行的所有存储脚本（包括全局和本地两种脚本）的名称。

连接到目标数据库和恢复目录，然后使用 REPLACE SCRIPT 命令，就可更新存储脚本。如果脚本不存在，RMAN 会创建脚本。

要从恢复目录中删除某一存储脚本，请连接到恢复目录和某个目标数据库，然后使用 DELETE SCRIPT 命令。

备份恢复目录



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

备份恢复目录

恢复目录是一个 Oracle DB，因此与任何数据库一样，需要进行备份。Oracle 建议使用 RMAN 进行备份，当然，应将控制文件而不是恢复目录用作 RMAN 资料档案库。千万不要将包含数据库的 RMAN 资料档案库的恢复目录存储在与目标数据库相同的数据库中，或与目标数据库相同的磁盘上。只有独立于要保护的数据时，恢复目录才有效。

配置控制文件自动备份，以便每次备份包含恢复目录时，都备份控制文件。任何时候在目标数据库中创建备份时，都可随后备份恢复目录。这样做可保护最新备份记录。

下面概要介绍了如何针对恢复目录配置备份和恢复环境：

- 在 ARCHIVELOG 模式下运行恢复目录。
- 将保留策略的 REDUNDANCY 设置为大于 1 的值。
- 将恢复目录备份到磁盘和磁带。
- 要创建备份，请使用 BACKUP DATABASE PLUS ARCHIVELOG 命令。
- 使用控制文件 (NOCATALOG) 而不是其它恢复目录作为 RMAN 资料档案库。
- 将控制文件自动备份配置为 ON。

重新创建无法恢复的恢复目录

要重新创建丢失的恢复目录中的部分内容，请使用以下命令：

- **RESYNC CATALOG 命令：**使用来自目标数据库控制文件或控制文件副本的任何 RMAN 资料档案库信息更新恢复目录
- **CATALOG START WITH 命令：**重新将任何可用备份列入目录



版权所有 © 2011, Oracle。保留所有权利。

重新创建无法恢复的恢复目录

如果恢复目录数据库已丢失或已损坏，并且通过正常 Oracle 恢复过程无法恢复恢复目录数据库时，就必须重新创建恢复目录。

可以使用以下命令重新填充恢复目录的部分内容：

- **RESYNC CATALOG：**通过此命令，使用来自目标数据库控制文件或控制文件副本的任何 RMAN 资料档案库信息更新恢复目录。请注意，来自因时间太早而从控制文件中清除的控制文件记录的元数据会丢失。
- **CATALOG START WITH...：**使用此命令可重新将所有可用备份列入目录。

导出和导入恢复目录

使用导出和导入实用程序或数据泵实用程序可执行以下操作：

- 将恢复目录从一个数据库移至另一个数据库
- 创建 RMAN 恢复目录的逻辑备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

导出和导入恢复目录

可以使用导出和导入功能将恢复目录从一个数据库移至另一个数据库。

还可以创建恢复目录的导出文件作为逻辑备份。

要从一个数据库中导出恢复目录然后将其导入到另一个数据库中，请执行以下步骤：

1. 使用 Oracle 导出实用程序之一从数据库中导出目录数据。
2. 在导出到的数据库上创建一个恢复目录用户，然后授予该用户必要的权限。
3. 使用对应的导入实用程序将目录数据导入到在步骤 2 中创建的方案。

请勿在将目录导入到数据库之前或之后执行 CREATE CATALOG 命令。导入操作会在第二个数据库中创建目录。

注：通过使用导出和导入或数据泵，同时使用逻辑方法，可以备份恢复目录，并将其作为可移动表空间移动至其它数据库。

升级和删除恢复目录

要将恢复目录升级到 RMAN 客户机所需的版本，请使用
UPGRADE CATALOG 命令：

```
UPGRADE CATALOG;
```

要删除恢复目录方案，请使用 DROP CATALOG 命令：

```
DROP CATALOG;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

升级和删除恢复目录

如果使用的恢复目录版本低于 RMAN 客户机需要的版本，则必须通过执行 UPGRADE CATALOG 命令来升级它。

- 要安装新的恢复目录方案，恢复目录用户必须具有 CREATE TYPE 权限。
- 必须连接到目录数据库，而且目录数据库必须处于打开状态。不必连接到目标数据库。
- 必须再次输入 UPGRADE 命令来确认升级。如果恢复目录的版本已高于 RMAN 执行文件所需的版本，您将收到错误消息。但是，如果恢复目录是最新版本，RMAN 将允许此命令运行，所以可根据需要重新创建程序包。
- RMAN 会在消息日志中显示升级期间生成的所有错误消息。

如果不再需要保留恢复目录，可以使用 DROP CATALOG 命令从表空间中删除恢复目录方案。删除目录时，会删除目录中注册的所有目标数据库的备份恢复目录记录。

- 只能在 RMAN 提示符下执行此命令。
- 必须通过 CATALOG 命令行选项或 CONNECT CATALOG 命令连接到恢复目录数据库。目录数据库必须处于打开状态。不必连接到目标数据库。
- 请输入命令两次，确认要删除方案。

IMPORT CATALOG 命令

1. 连接到目标恢复目录:

```
CONNECT CATALOG cat111/oracle@destdb;
```

2. 为所有注册的数据库导入元数据:

```
IMPORT CATALOG cat102/oracle@srcdb;
```

3. 为两个注册的数据库导入元数据:

```
IMPORT CATALOG cat92/oracle@catdb DBID=1423241, 1423242;
```

4. 从多个目录导入元数据:

```
IMPORT CATALOG cat102/rman@srcdb;
IMPORT CATALOG cat101/rman@srcdb;
IMPORT CATALOG cat92/rman@srcdb NO UNREGISTER;
```

必须等于 RMAN 可
执行文件的当前版本

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

IMPORT CATALOG 命令

使用 IMPORT CATALOG 命令可将元数据从一个恢复目录方案导入至其它目录方案中。如果创建了不同版本的目录方案来存储多个目标数据库的元数据, 则使用此命令可以为所有数据库维护单个目录方案。

```
IMPORT CATALOG <connectStringSpec>
[DBID = <dbid> [, <dbid>, ...]]
[DB_NAME=<dbname>[, <dbname>, ...]]
[ NO UNREGISTER ];
```

<connectStringSpec> 是源恢复目录连接字符串。源恢复目录方案的版本必须等于 RMAN 可执行文件的当前版本。如果需要, 将源目录升级到当前 RMAN 本版。

DBID: 您可以指定数据库 ID 的列表, 数据库 ID 的元数据应从源目录方案导入。未指定列表时, RMAN 将所有数据库 ID 的元数据从源目录方案合并到目标目录方案中。如果已在恢复目录方案中注册了合并元数据的数据库, RMAN 就会发出错误消息。

IMPORT CATALOG 命令 (续)

DB_NAME: 可以指定应导入其元数据的数据库的名称列表。如果数据库名称不明确, RMAN 就会发出错误消息。

NO UNREGISTER: 默认情况下, 导入的数据库 ID 在成功导入后从源恢复目录方案中注销。使用 NO UNREGISTER 选项, 可以强制 RMAN 将导入的数据库 ID 保留在源目录方案中。

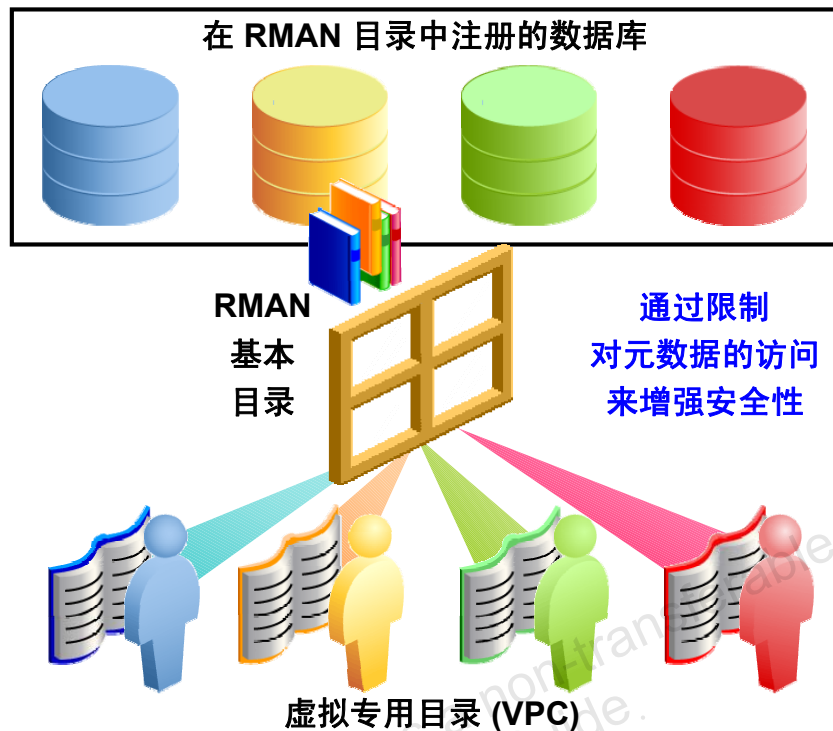
导入示例 (续)

1. RMAN 必须连接到需要将目录数据导入到的目标恢复目录, 例如 cat111 方案。这是幻灯片中所有给定示例中的第一步。
注: 源恢复目录方案的版本必须等于 RMAN 可执行文件的当前版本。导入早期版本的目录之前, 必须将早期版本升级至 RMAN 可执行文件的版本。
2. 在本例中, cat102 用户在 srcdb 数据库中拥有一个 RMAN 目录。您希望 RMAN 导入所有注册的数据库并在源目录中注销这些数据库。
3. cat92 用户在 srcdb 数据库中拥有一个 RMAN 目录。您希望 RMAN 导入 DBID 为 1423241 和 1423242 的数据库, 并在源目录中注销这些数据库。
4. srcdb 数据库包含三个不同的恢复目录。RMAN 将所有数据库 ID (已在这些目录中注册) 的元数据导入到 destdb 数据库中的 cat111 方案。除在 cat92 方案中注册的数据库之外, 所有导入的目标数据库将从其源目录中注销。

有关使用的其它详细资料

- 确保没有目标数据库既在源目录方案中注册, 又在目标目录方案中注册。如果某个目标数据库在这两个方案中都注册了, 请将此数据库从源目录中注销, 然后重试导入。
- 如果导入过程中操作失败, 则将回退导入。从不存在部分导入的情况。
- 源目录方案和目标目录方案中的存储脚本名称发生冲突时, RMAN 会重命名源目录方案的存储脚本。

创建和使用虚拟专用目录



ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

创建和使用虚拟专用目录

使用此功能可以合并 RMAN 资料档案库并保持职责分离，这是一项基本安全要求。

RMAN 目录已被增强，可用于创建数据库组 and 用户组的虚拟专用 RMAN 目录。该目录的所有者将创建基本目录并为虚拟目录的所有者授予 RECOVERY_CATALOG_OWNER 权限。目录的所有者可向虚拟目录的所有者授予访问已注册数据库的权限或 REGISTER 权限。然后，虚拟目录所有者可连接到特定目标的目录，或者注册一个目标数据库。完成配置后，VPC 所有者可像使用标准基本目录一样使用虚拟专用目录。

作为目录所有者，您可以访问该目录中所有已注册的数据库信息。您可以列出使用 SQL*Plus 命令注册的所有数据库：

```
SELECT DISTINCT db_name FROM DBINC;
```

作为虚拟目录所有者，您只能查看授予了您访问权限的数据库。

注：如果目录所有者未被授予对目标数据库的 SYSDBA 或 SYSOPER 权限，则无法执行大多数 RMAN 操作。

使用 RMAN 虚拟专用目录

1. 创建 RMAN 基本目录:

```
RMAN> CONNECT CATALOG catowner/oracle@catdb
RMAN> CREATE CATALOG;
```

2. 将 RECOVERY_CATALOG_OWNER 授予给 VPC 所有者:

```
SQL> CONNECT SYS/oracle@catdb AS SYSDBA
SQL> GRANT RECOVERY_CATALOG_OWNER to vpcowner;
```

3a. 将 REGISTER 授予给 VPC 所有者:

```
RMAN> CONNECT CATALOG catowner/oracle@catdb
RMAN> GRANT REGISTER DATABASE TO vpcowner;
```

3b. 或将 CATALOG FOR DATABASE 授予给 VPC 所有者:

```
RMAN>GRANT CATALOG FOR DATABASE db10g TO vpcowner;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 RMAN 虚拟专用目录

可以为数据库组 and 用户组创建虚拟专用 RMAN 目录。

1. 目录所有者将创建基本目录。
2. 目录数据库的 DBA 可以创建拥有虚拟专用目录 (VPC) 的用户, 并授予其 RECOVERY_CATALOG_OWNER 权限。
3. 基本目录的所有者可为 VPC 所有者授予访问先前注册的数据库的权限或授予 REGISTER 权限。GRANT CATALOG 命令如下:

```
GRANT CATALOG FOR DATABASE prod1, prod2 TO vpcowner;
```

GRANT REGISTER 命令如下:

```
GRANT REGISTER DATABASE TO vpcowner;
```

然后, 虚拟目录所有者可连接到特定目标的目录, 或者注册一个目标数据库。配置完 VPC 之后, VPC 所有者可像使用标准基本目录一样使用该目录。

使用 RMAN 虚拟专用目录

4a. 为 11g 客户机创建虚拟目录:

```
RMAN> CONNECT CATALOG vpcowner/oracle@catdb
RMAN> CREATE VIRTUAL CATALOG;
```

4b. 或者为 11g 之前的客户机创建虚拟目录:

```
SQL> CONNECT vpcowner/oracle@catdb
SQL> exec catowner.dbms_rcvcat.create_virtual_catalog;
```

5. 在目录中注册新数据库:

```
RMAN> CONNECT TARGET / CATALOG vpcowner/oracle@catdb
RMAN> REGISTER DATABASE;
```

6. 使用虚拟目录:

```
RMAN> CONNECT TARGET / CATALOG vpcowner/oracle@catdb;
RMAN> BACKUP DATABASE;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 RMAN 虚拟专用目录 (续)

4. 创建虚拟专用目录。

a. 如果目标数据库是 Oracle Database 11g 数据库并且 RMAN 客户机是 11g 客户机, 则可使用 RMAN 命令:

```
CREATE VIRTUAL CATALOG;
```

b. 如果目标数据库是 Oracle Database 10g 发行版 2 或更早版本 (使用兼容客户机), 则必须从 SQL*Plus 执行提供的过程:

```
BASE_CATALOG_OWNER.DBMS_RCVCAT.CREATE_VIRTUAL_CATALOG;
```

5. 使用 VPC 所有者登录名连接到该目录, 将其作为常规目录使用。

6. 虚拟目录所有者只能查看其有访问权限的那些数据库。对于大多数 RMAN 操作, 您还需要针对目标数据库的 SYSDBA 或 SYSOPER 权限。

恢复目录概要

管理恢复目录：

1. 创建恢复目录。
2. 在恢复目录中注册目标数据库。
3. 如果需要，使用新的 `IMPORT CATALOG` 命令合并恢复目录。
4. 如果需要，将所有旧备份列入目录。
5. 如果需要，为特定用户创建新的虚拟恢复目录。
6. 保护恢复目录。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复目录概要

管理恢复目录的基本工作流不是新的。但是，进行了增强，其中两项重要功能发生了变化：合并 RMAN 资料档案库和虚拟专用目录，这样就可以职责分离。

1. 创建恢复目录。
2. 在恢复目录中注册目标数据库。通过此步骤，RMAN 可以将目标数据库的元数据存储在恢复目录中。
3. 如果需要，也可使用 `IMPORT CATALOG` 命令合并恢复目录。
4. 如果需要，将所有旧备份列入目录，旧备份的记录不再存储在目标控制文件中。
5. 如果需要，为特定用户创建虚拟恢复目录并确定用户具有访问权限的元数据。
6. 通过将恢复目录包含在备份和恢复策略中来保护恢复目录。

恢复目录概要（续）

恢复目录中包含了有关每个已注册目标数据库的 RMAN 操作的元数据。恢复目录包含以下元数据类型：

- 数据文件和归档重做日志备份集以及备份片段
- 数据文件副本
- 归档重做日志及其副本
- 目标数据库中的表空间和数据文件
- 存储脚本，它们是由用户创建的 RMAN 命令序列
- 永久性 RMAN 配置设置

在恢复目录中注册目标数据库以供 RMAN 使用的过程称为注册。建议在单个恢复目录中注册所有目标数据库。例如，可以在 catdb 数据库的 catowner 方案所拥有的单个目录中注册 prod1、prod2 和 prod3 数据库。

集中恢复目录（也称为基本恢复目录）的所有者可以授予或撤销其他数据库用户对该目录的受限访问权限。所有元数据都存储在基本目录方案中。

每个受限用户都具有对其各自元数据的完整读写权限，这称为虚拟专用目录。

恢复目录从每个已注册目标数据库的控制文件中获取重要的 RMAN 元数据。重新同步恢复目录确保了 RMAN 从控制文件获取的元数据是当前元数据。

可以作为命令文件的替代方式使用存储脚本来管理经常使用的 RMAN 命令序列。脚本存储在恢复目录中，而不是文件系统中。本地存储脚本与创建脚本时 RMAN 连接到的目标数据库相关联，只有连接到该目标数据库时才能执行本地存储脚本。全局存储脚本可针对在恢复目录中注册的任何数据库运行。

可在使用或已使用不同数据库版本的环境中使用恢复目录。因此，您的环境可具有不同版本的 RMAN 客户机、恢复目录数据库、恢复目录方案和目标数据库。**为便于管理，可将一个恢复目录（或此目录中特定数据库的元数据）合并到另一个恢复目录中。**

测验

请选择有关 Oracle 恢复目录的所有正确陈述：

1. 使用恢复目录，您可以存储更长的备份历史记录，比基于控制文件的资料档案库所存储的历史记录还要长。
2. Oracle 建议您对所有数据库一律使用恢复目录。
3. 为了在 EM 中使用恢复目录进行备份和恢复的相关操作，必须使用 EM 注册方法。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1、3

测验

RMAN 目录方案可以使用数据泵导出实用程序进行备份。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1

小结

在本课中，您应该已经学会：

- 对恢复目录和 RMAN 资料档案库控制文件的使用进行比较
- 创建和配置恢复目录
- 在恢复目录中注册数据库
- 同步恢复目录
- 使用 RMAN 存储脚本
- 备份恢复目录
- 创建虚拟专用目录

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 3 概览：使用 RMAN 恢复目录

本练习包含以下主题：

- 使用 dbca 实用程序创建 RCAT 恢复目录实例
- 创建恢复目录所有者并授予权限
- 在 RMAN 中创建恢复目录
- 注册 ORCL 数据库
- 备份 RCAT 恢复目录

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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

配置备份设置



ORACLE®

版权所有 © 2011, 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：

- 使用 Oracle Enterprise Manager 配置备份设置
- 启用控制文件自动备份
- 配置备份目标
- 为磁带目标分配通道
- 配置备份优化
- 创建压缩备份
- 创建加密备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置 RMAN 的永久性设置

- RMAN 有预设的默认配置设置。
- 使用 CONFIGURE 命令可以：
 - 配置自动通道
 - 指定备份保留策略
 - 指定要创建的备份副本数
 - 将默认备份类型设置为 BACKUPSET 或 COPY
 - 限制备份片段的大小
 - 在备份中排除表空间
 - 启用和禁用备份优化
 - 配置控制文件的自动备份
 - 定义归档日志删除策略
 - 指定设备的并行度
 - 设置用于备份的加密参数和压缩参数

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置 RMAN 的永久性设置

为了简化使用 RMAN 进行备份和恢复的过程，可以使用 RMAN 为每个目标数据库设置一些永久性配置设置。这些设置可控制 RMAN 行为的多个方面。可以保存永久性配置信息，如通道参数、并行度和 RMAN 资料档案库中的默认设备类型。这些配置设置始终存储在控制文件和恢复目录数据库中（如果存在）。

这些设置都提供了默认值，您可以立即使用 RMAN。但是，当制定更高级的备份和恢复策略时，可能要更改这些设置才能实施该策略。使用 CONFIGURE 命令可为 RMAN 备份、还原、复制和维护作业配置永久性设置。这些设置在所有 RMAN 会话中均有效，直到清除或更改了配置为止。

注：使用 SET 命令在 RMAN 作业（或会话）中更改的配置的有效时间仅为作业（或会话）的持续时间。

EM 注：通过 Oracle Enterprise Manager 界面使用 RMAN 时，情况也是如此。备份设置为所发生的所有备份提供默认设置。在创建某个备份时，可以为该特定备份覆盖其中的一些设置。

查看永久性设置

要检查数据库的永久性 RMAN 设置：

- 如果仅连接到了目标，请在 RMAN 提示符下输入
SHOW ALL。

或：

- 如果已登录到目标数据库实例，请查询
V\$RMAN_CONFIGURATION 视图。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

查看永久性设置

连接至目标并输入 SHOW ALL 命令时，或登录到 SQL*Plus 并查询 V\$RMAN_CONFIGURATION 视图时，可查看 RMAN 永久性设置。

示例：

```
SQL> select * from V$RMAN_CONFIGURATION
2 /
```

CONF#	NAME	VALUE
1	CONTROLFILE AUTOBACKUP	ON
2	CHANNEL	DEVICE TYPE 'SBT_TAPE' PARMS 'SBT_LIBRARY=oracle.disksbt,ENV=(BACKUP_DIR=/tape)'

控制文件自动备份

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

Backup Settings	
Device	Backup Set
<p>Backup Policy</p> <p><input checked="" type="checkbox"/> Automatically backup the control file and server parameter file (SPFILE) with every backup and database structural change</p> <p>Autobackup Disk Location <input type="text"/></p> <p><small>An existing directory or diskgroup name where the control file and server parameter file will be backed up. If you do not specify a location, the files will be backed up to the flash recovery area location.</small></p>	

最佳做法提示：Oracle 建议您启用控制文件自动备份。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

控制文件自动备份

您可以使用 Oracle Enterprise Manager 指定实例的备份设置。在“Database（数据库）”主页中，导航到“Availability > Backup Settings（可用性 > 备份设置）”。

要轻松地所有控制文件副本丢失中恢复过来，应将 RMAN 配置为自动备份控制文件。控制文件的自动备份与备份命令明确请求的对当前控制文件的备份无关。如果在 NOCATALOG 模式下运行 RMAN，则强烈建议您激活控制文件自动备份功能。否则，如果丢失了控制文件，就可能无法恢复数据库。

要配置控制文件自动备份，请使用 Oracle Enterprise Manager 或使用以下 RMAN 命令修改数据库的备份策略：

```
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

默认情况下，控制文件自动备份处于禁用状态。如果启用了控制文件自动备份，则 RMAN 在出现以下情况时会自动备份控制文件和当前服务器参数文件（如果该文件用来启动数据库）：

- 运行脚本结束时
- 在 RMAN 资料档案库中记录了成功的备份时
- 当数据库发生结构性更改时，Oracle 内核本身会备份（例如，在发生影响控制文件内容的 DDL 操作后）

控制文件自动备份（续）

对于所有设备类型，控制文件自动备份文件名都有一个默认格式 %F，因此 RMAN 不需要资料档案库就可以推断出文件位置，并还原该文件。该变量格式会转换为 c-FFFFFFFFII-YYYYMMDD-QQ，其中：

- FFFFFFFFFI 代表 DBID
- YYYYMMDD 是生成备份那天的时间戳
- QQ 是十六进制序列，从 00 开始，最大值为 FF

您可以使用 CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE *type* TO '*string*' 命令更改默认格式。字符串的值必须包含替代变量 %F，并且不能包含其它替代变量。例如：

```
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT  
FOR DEVICE TYPE DISK TO '/u01/oradata/cf_ORCL_auto_%F';
```

除非另外指明，否则控制文件自动备份存储在快速恢复区中。

使用控制文件自动备份，即使无法访问当前控制文件、恢复目录和服务器参数文件，RMAN 仍可恢复数据库。因为用于存储自动备份的路径遵从已知格式，所以 RMAN 可从自动备份中搜索并还原服务器参数文件或控制文件。

管理永久性设置

- 使用流入及流出设备的多个数据流：

```
RMAN> CONFIGURE DEVICE TYPE sbt PARALLELISM 3;
```

- 使用 SHOW 命令列出当前设置：

```
RMAN> SHOW CONTROLFILE AUTOBACKUP FORMAT;  
RMAN> SHOW EXCLUDE;  
RMAN> SHOW ALL;
```

- 使用 CONFIGURE 命令的 CLEAR 选项可将任何永久性设置重置为其默认值：

```
RMAN> CONFIGURE BACKUP OPTIMIZATION CLEAR;  
RMAN> CONFIGURE MAXSETSIZE CLEAR;  
RMAN> CONFIGURE DEFAULT DEVICE TYPE CLEAR;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理永久性设置

并行度是可用于对设备进行读取和写入的数据流的数量。当设备由 RMAN 使用时，这可以有效地分配通道的数量。例如，如果某个介质管理器有两个磁带机可用，则并行度为 2 时使用该介质管理器 BACKUP 命令可以同时使用两个磁带驱动器。要将某个备份分布在多个磁盘中时，磁盘设备类型的并行度也很有用。

可以使用如下所示的 PARALLELISM 子句指定设备上使用的并行度：

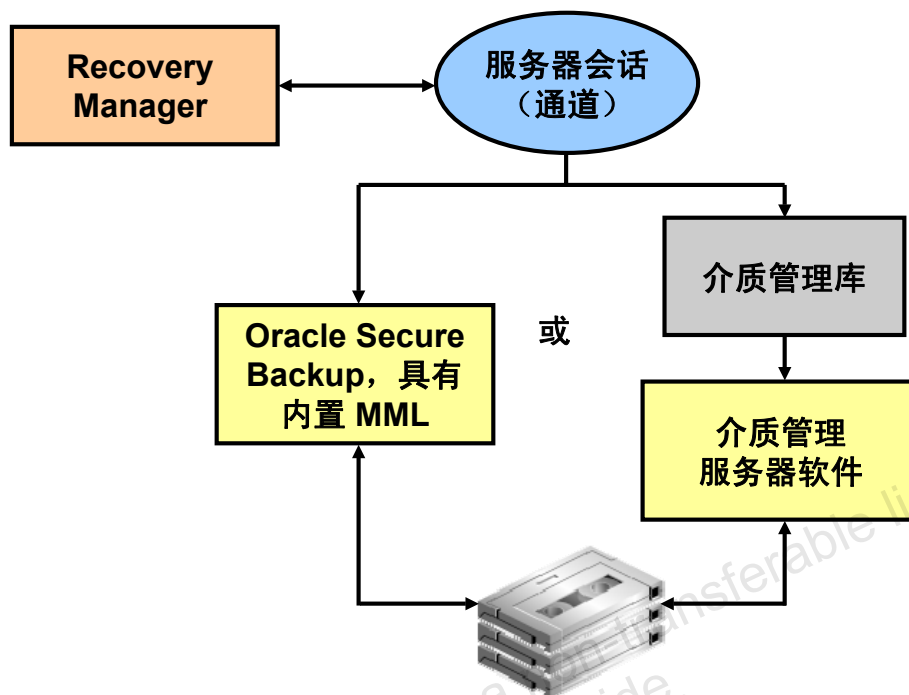
```
CONFIGURE DEVICE TYPE <device> PARALLELISM <n>
```

其中 <n> 是并行度值。

使用 RMAN 的 SHOW 命令可查看 RMAN 配置设置。如果连接到目标数据库后执行 SHOW ALL，则仅显示特定于节点的配置和数据库配置。

通过执行带 CLEAR 选项的 CONFIGURE 命令，可以将同一 CONFIGURE 命令设置的值恢复为其默认值。

使用介质管理器



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用介质管理器

要将磁带存储用于数据库备份，RMAN 需要使用 Oracle Secure Backup 或介质管理器。

介质管理器是一个用于加载、标注和卸载顺序介质（如磁带机）以备份、还原和恢复数据的实用程序。Oracle DB Server 调用介质管理库 (MML) 软件例程在由介质管理器控制的介质之间备份和还原数据文件。

请注意，Oracle DB Server 在备份到磁盘时并不需要连接到 MML 软件。

Oracle Backup Solutions Program (BSP) 提供了一系列符合 Oracle MML 规范的介质管理产品。通过与 MML 接口兼容的软件，Oracle DB 会话可将数据备份到介质管理器，并可请求介质管理器还原备份。请与介质供应商联系，确认该供应商是否为 Oracle BSP 的成员。

必须先安装介质管理器软件并确保 RMAN 可与该软件进行通信，然后才能使用 RMAN 与介质管理器。在介质管理器供应商的软件文档中可找到此过程的说明。

使用介质管理器（续）

请根据要安装的产品执行以下基本步骤：

1. 在目标主机或生产网络中安装并配置介质管理软件。此阶段不需要集成 RMAN。
2. 请确保在目标数据库主机上可对操作系统文件执行非 RMAN 备份。此步骤有助于以后的问题排除。请参阅介质管理文档，了解如何将文件备份到介质管理器。
3. 获取并安装第三方介质管理模块，以便与 Oracle DB 集成。该模块必须包含访问介质管理器时由 Oracle DB Server 加载的库。

使用介质管理器备份和还原操作

以下 Recovery Manager 脚本执行的操作是将数据文件备份到由介质管理器控制的磁带机：

```
run {  
  # Allocating a channel of type 'sbt' for serial device  
  ALLOCATE CHANNEL ch1 DEVICE TYPE sbt;  
  BACKUP DATAFILE 3;  
}
```

Recovery Manager 执行该命令时，会将备份请求发送到执行备份的 Oracle DB 会话。

Oracle DB 会话将输出通道标识为介质管理设备，然后请求介质管理器加载磁带并写入输出。

介质管理器会给磁带加上标签并跟踪每一磁带上文件的文件名。介质管理器还会处理还原操作。还原文件时，会执行以下步骤：

1. Oracle DB Server 请求还原某个特定文件。
2. 介质管理器确定包含文件的磁带并读取磁带。
3. 介质管理器将信息传送回 Oracle DB 会话。
4. Oracle DB Server 将文件写入到磁盘。

指定备份目标

备份可写入到：

- 磁盘目录

```
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO DISK;
```

- 磁带（使用 Oracle Secure Backup）
- 介质管理库
 - 磁带

```
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO TAPE;
```

- 磁盘或磁带（使用代理副本）
- 快速恢复区：专门为备份和恢复以及闪回数据库而留出的磁盘区
 - 定义位置和大小
 - 文件会根据需要自动保留和删除

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

指定备份目标

备份可以写入到指定的磁盘目录、介质管理库 (MML) 或快速恢复区。指定磁盘目录或快速恢复区意味着备份会移到硬盘介质。通常，为了保持磁盘空间可用，会定期通过介质管理接口将备份脱机移到磁带中。只要磁盘目录已存在，任何磁盘目录都可以指定为备份目标。

介质管理库可用于将文件复制到磁带机，或执行代理副本。代理副本是要求使用 MML 在磁盘或磁带机上创建文件副本的位置。MML 必须能提供代理副本服务才能创建副本。

如果设置了快速恢复区，则可简化许多备份和恢复任务。Oracle DB 会自动指定文件，在空间紧缺时会删除过时的文件。

要指定将备份写入到磁盘，使用幻灯片中的第一个命令。

随后，当创建备份时，如果使用 FORMAT 关键字（指定备份的磁盘目录位置），则会将备份写入到该位置。如果配置了快速恢复区，则会备份到该位置；否则，备份会写入到平台特定的默认位置。

要指定要使用某个磁带机。请使用幻灯片中的第二个命令。

注：有关 Oracle Secure Backup 的详细信息，请参阅《Oracle Secure Backup Administrator's Guide》。

配置和分配通道

- 使用 CONFIGURE 命令配置自动通道:

```
RMAN> CONFIGURE DEVICE TYPE sbt;  
RMAN> CONFIGURE DEFAULT DEVICE TYPE TO sbt;  
RMAN> CONFIGURE CHANNEL DEVICE TYPE sbt ...  
RMAN> BACKUP DATABASE;
```

- 在 RUN 块内使用 ALLOCATE CHANNEL 命令手动分配通道:

```
RMAN> RUN  
{  
  ALLOCATE CHANNEL ch1 DEVICE TYPE DISK;  
  BACKUP DATABASE PLUS ARCHIVELOG;  
}
```

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

配置和分配通道

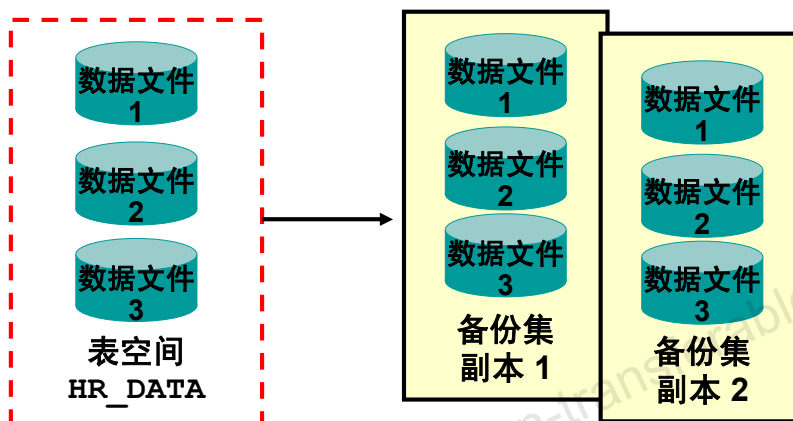
请从以下选项中进行选择来配置通道和执行备份:

- 使用 CONFIGURE 命令配置自动通道, 然后在 RMAN 提示符下或在 RUN 块中发出 BACKUP 命令。
- 在 RUN 块中使用 ALLOCATE CHANNEL 命令手动分配通道, 然后发出 BACKUP 命令。

创建双向备份集

要创建双向备份集，请使用：

- CONFIGURE ... BACKUP COPIES
- BACKUP ... COPIES



ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

创建双向备份集

RMAN 最多可同时为备份集创建四个副本，这些副本彼此完全一样。备份集的副本是备份集中所有备份片段的副本，每个副本都有一个唯一的副本编号（例如，0tcm8u2s_1_1 和 0tcm8u2s_1_2）。

多数情况下，创建双向备份集的最简便方法是使用 BACKUP ... COPIES 或 CONFIGURE ... BACKUP COPIES 创建双向备份集。对于 DISK 通道，在 FORMAT 选项中指定多个值，可将多个副本放到不同的物理磁盘中。对于 sbt 通道，如果使用支持 SBT API 版本 2 的介质管理器，则介质管理器会自动将每个副本放在一个单独的介质上（例如，单独的磁带）。

注：系统备份到磁带 (SBT) API 是为介质管理库 (MML) 开发人员定义的接口，以便他们可以提供与 RMAN 进行通信的 MML。

请注意，不能在快速恢复区中创建双向备份集，而且双向备份只适用于备份集，不适用于映像副本。如果在创建映像副本备份时指定了 BACKUP ... COPIES 选项，则您会收到错误。对于映像副本备份来说，会忽略 CONFIGURE ... BACKUP COPIES 设置。

双向备份集通常用于磁带备份。

使用 CONFIGURE BACKUP COPIES 创建双向备份集

```

RMAN> CONFIGURE ARCHIVELOG BACKUP COPIES
2> FOR DEVICE TYPE sbt TO 2;
RMAN> CONFIGURE DATAFILE BACKUP COPIES
2> FOR DEVICE TYPE sbt TO 2;
RMAN> BACKUP DATABASE PLUS ARCHIVELOG;
RMAN> BACKUP DEVICE TYPE DISK AS COPY DATABASE;

```

创建备份到两个不同磁带的两个副本。

不受 COPIES 配置设置的影响。只创建一个备份到磁盘的副本。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 CONFIGURE BACKUP COPIES 创建双向备份集

使用 CONFIGURE ... BACKUP COPIES 命令可指定要在指定设备类型上创建的完全一样的备份集的数量。使用 BACKUP BACKUPSET 命令进行备份时，此设置适用于除控制文件自动备份之外的所有备份（因为控制文件的自动备份始终生成一个副本）和备份集。

注：必须已配置了自动通道。

要使用 CONFIGURE BACKUP COPIES 创建双向备份集，请执行以下步骤：

1. 在所需设备类型上，配置数据文件和归档重做日志文件的副本数。
2. 执行 BACKUP 命令。
3. 发出 LIST BACKUP 命令来验证备份。

注：最后一个 BACKUP 命令不受 COPIES 配置设置的影响。该命令将创建备份到磁盘的单个副本。

备份优化

- 跳过已备份的文件
- 在以下情况下使用：
 - 启用了备份优化

```
RMAN> CONFIGURE BACKUP OPTIMIZATION ON;
```

- 执行了 BACKUP DATABASE、带 ALL 或 LIKE 选项的 BACKUP ARCHIVELOG 或 BACKUP BACKUPSET ALL 命令
- 只分配了一种类型的通道
- 使用 FORCE 选项可以被覆盖

```
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET ALL FORCE;
```

- 始终用于 RECOVERY AREA、DB_RECOVERY_FILE_DEST 和 RECOVERY FILES 备份选项

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

备份优化

如果启用了备份优化，当完全一样的文件已备份到指定的设备类型时，BACKUP 命令就会跳过这些文件。

如果 RMAN 确定某个文件是完全一样的且已备份，则该文件就是可以跳过的候选文件。但是，RMAN 会通过执行进一步的检查来确定是否跳过该文件，因为 RMAN 用于确定指定设备类型上是否存在充足备份的算法会考虑保留策略和双重备份功能这两个因素。

有关 RMAN 用于确定文件是否完全一样的标准以及备份优化算法的详细信息，请参阅《Oracle Database Backup and Recovery User's Guide》。

在 Oracle Enterprise Manager 中的“Backup Settings（备份设置）”页中，或者通过发出 CONFIGURE BACKUP OPTIMIZATION ON 命令均可启用备份优化。默认情况下，备份优化处于禁用状态。

对于 BACKUP RECOVERY AREA | DB_RECOVERY_FILE_DEST 和 BACKUP RECOVERY FILES 命令系统会自动启用备份优化。

配置备份优化（续）

要覆盖备份优化并备份所有文件（无论是否已更改），请在 BACKUP 命令中指定 FORCE 选项，如下例所示：

```
BACKUP DEVICE TYPE sbt BACKUPSET ALL FORCE;
```

请注意，FORCE 选项并不适用于恢复区中的文件。

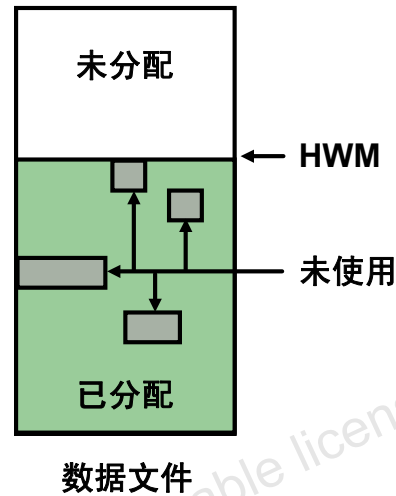
使用 Oracle Enterprise Manager 或通过发出下列命令可永久地禁用备份优化：

```
CONFIGURE BACKUP OPTIMIZATION OFF;
```

通过压缩未使用的块节省备份空间

在某些类型的备份操作过程中，可跳过以下块：

- 未分配的块：这些块高于数据文件的高水位标记 (HWM)。
- 未使用的块：这些块已分配但已不再属于段。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

通过压缩未使用的块节省备份空间

执行某些类型的备份时，RMAN 可跳过一些块。可跳过未分配的块。这些块尚未分配；它们高于 HWM。此外，当符合以下条件时，还可以跳过某些已不再属于段的已分配块（不再使用）：

- 未定义保证还原点。
- 数据文件仅包含本地管理的表空间的数据。
- 数据文件是作为完全备份或 0 级增量备份的一部分备份到备份集的。
- 备份将保存到磁盘中，或者 Oracle Secure Backup 是介质管理器。

压缩备份

RMAN 可对生成的任何备份集执行二进制压缩。

- 除了压缩未使用的块以外，还可以执行二进制压缩。
- 可用的压缩算法为：HIGH、MEDIUM、LOW 和 BASIC。
- DBA 无需执行额外步骤便可还原压缩的备份。

```
CONFIGURE COMPRESSION ALGORITHM 'HIGH/MEDIUM/LOW/BASIC'
```

```
run {
SET COMPRESSION ALGORITHM 'HIGH/MEDIUM/LOW/BASIC';
..
}
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

压缩备份

不对事务处理恢复不需要的还原数据（例如，针对已提交的事务处理）进行备份。优点是从总体上减少了备份时间和存储空间，因为它不备份应用于已提交的事务处理的还原数据。自动启用此优化。

压缩未使用的块可减少写入到备份中的块数（以及备份时间），而二进制压缩可使用算法压缩写入的数据。可用的压缩算法为 HIGH、MEDIUM、LOW 和 BASIC。如果为特定的备份设备指定它，请在 BACKUP TYPE TO 子句后使用 COMPRESSED 关键字。

无需执行任何其它步骤便可还原压缩的备份。但应注意的是，压缩和解压缩操作需要占用 CPU 资源。因此，创建和还原压缩的备份可能需要更长的时间并需要占用更多的系统资源。

选择算法时，应考虑磁盘空间以及动态系统资源（如 CPU 和内存等）。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

使用 RMAN 备份压缩

压缩比率或压缩级别	注意事项	需要高级压缩选件
LOW	时间最短。最适用于解决备份：CPU 资源。	✓
MEDIUM	时间短。很好兼顾 CPU 使用率和压缩比率。	✓
HIGH	最佳压缩比率，其代价是高 CPU 消耗。最适用于解决备份约束：网络。	✓
BASIC	时间一般。压缩比率与使用 MEDIUM 时类似，其代价是额外 CPU 使用。 MEDIUM 与 HIGH 之间的压缩比率。	



版权所有 © 2011, Oracle。保留所有权利。

使用 RMAN 备份压缩

如幻灯片所示，备份集的二进制压缩由算法设置支持。除了 BASIC 外，所有模式都需要 Oracle 高级压缩数据库选件。

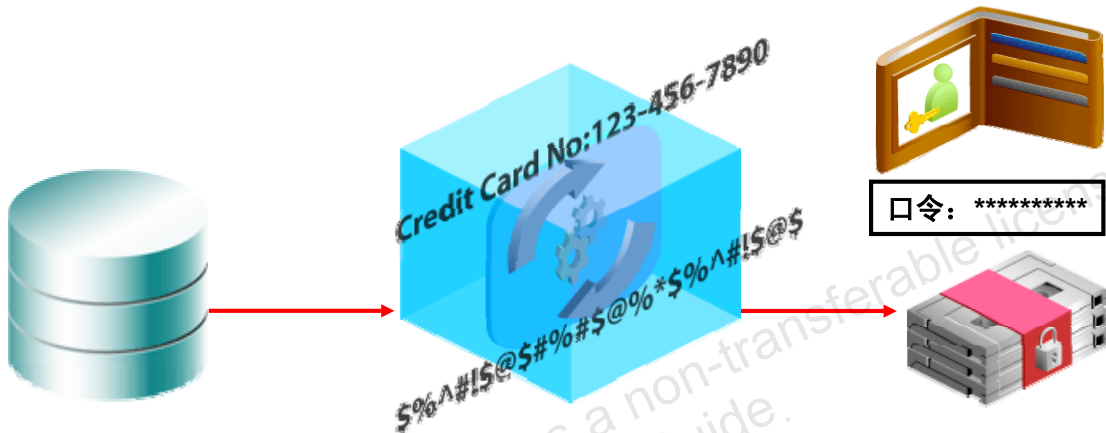
由于各种压缩级别的性能取决于数据库中数据的性质、网络配置、系统资源以及计算机系统类型和相应功能，Oracle Corporation 无法将可应用的性能统计信息全部提供说明。要找到最适合您的级别，需要考虑您系统在带宽到 CPU 的平衡性，以及 CPU 的实际速度。强烈建议您在您的环境中针对数据使用不同的压缩级别运行测试。要确保备份集压缩级别可以满足您公司性能需求和任一适用服务级别要求的唯一方法是基于您自己的环境、网络通信量（工作量）和数据集来选择压缩级别。

可以使用以下级别或压缩比率：

- LOW：此级别速度最快。与 MEDIUM 相比，它提供较少的压缩，但 CPU 使用率最低。（与 LZ0 压缩对应。）
- MEDIUM：此级别很好兼顾了 CPU 使用率和压缩比率。（与 ZLIB 压缩对应。）
- HIGH：此级别提供最佳压缩比率，但 CPU 消耗最大。（与 GZIP 压缩对应。）
- BASIC：与 BZIP2（10g 类型压缩）对应。

加密备份

- 透明加密：使用 wallet（默认）
- 口令加密：使用口令（无 wallet）
- 双模式加密：
 - 同时使用透明加密和口令加密模式
 - 既可以使用透明模式也可以使用口令模式进行还原



版权所有 © 2011, Oracle. 保留所有权利。

加密备份

可使用以下三种方法之一对备份进行加密：

- **透明加密：**此方法使用 wallet，是默认模式。
- **口令加密：**此加密方法依赖于口令，不需要配置 wallet。若要还原，必须知道用于备份的口令。
- **双模式加密：**可同时使用透明加密和口令加密。若要还原，可使用透明模式或口令模式。如果通常将备份还原到本地站点，但偶尔也会将备份传送到其它站点，则这种类型的加密十分有用。

有关加密备份的详细资料，请参阅“Oracle Database 11g：安全性”课程。

测验

如何检查数据库的永久性 RMAN 设置？选择所有正确答案：

1. 如果仅连接到了目标，请在 RMAN 提示符下输入
SHOW ALL。
2. 在 SQL*Plus 会话中，使用 SHOW RMAN
CONFIGURATION 命令。
3. 如果仅连接到了恢复目录，请在 RMAN 提示符下输入
SHOW ALL。
4. 如果已登录到目标数据库实例，请查询
V\$RMAN_CONFIGURATION 视图。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1、4

测验

选择有关 RMAN 备份功能的正确陈述：

1. 无论是否已更改，备份 `FORCE` 都会覆盖备份优化并备份所有文件。
2. 永久性 RMAN 设置只用于一次性备份。
3. 并行度是流入和流出备份设备的可能数据流的数量。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1、3

小结

在本课中，您应该已经学会：

- 使用 Oracle Enterprise Manager 配置备份设置
- 启用控制文件自动备份
- 配置备份目标
- 为磁带目标分配通道
- 配置备份优化
- 创建压缩备份
- 创建加密备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 4 概览：配置备份规范

本练习包含以下主题：

- 配置 RMAN 永久性设置
- 配置控制文件的自动备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

配置备份规范

做法提示：由于 RMAN 命令的输出可能会很长，可以考虑使用 RMAN SPOOL LOG 命令将输出引至您指定的文件。

示例

```
RMAN> SPOOL LOG TO '/home/oracle/labs/my_lab_output.txt';
```

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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

5 使用 RMAN 创建备份

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

学完本课后，应能完成以下工作：

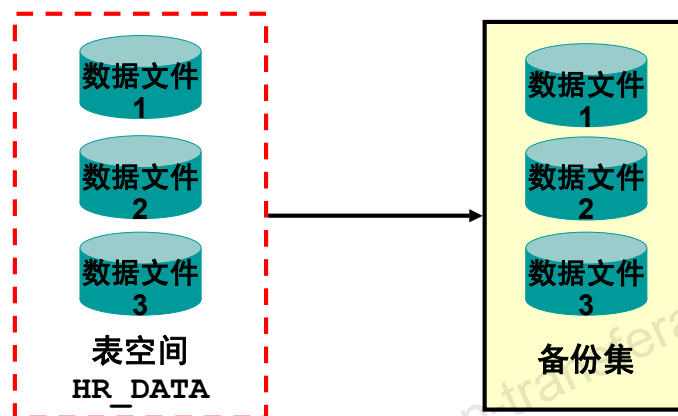
- 创建映像文件备份
- 创建整体数据库备份
- 创建完全数据库备份
- 启用快速增量备份
- 创建双向备份集
- 备份备份集
- 创建 RMAN 多部分备份
- 创建归档备份以用于长期保留
- 报告和维护备份

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

创建备份集

```
RMAN> BACKUP AS BACKUPSET  
2> FORMAT '/BACKUP/df_%d_%s_%p.bus'  
3> TABLESPACE hr_data;
```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

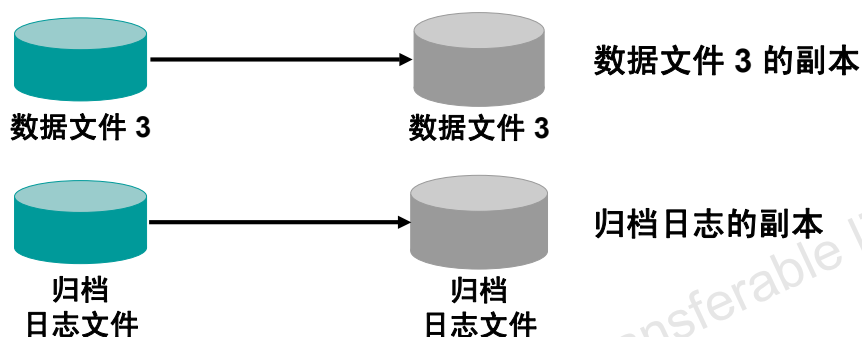
创建备份集

RMAN 可以按 RMAN 专用格式存储其备份，这种格式称为备份集。备份集是由被称为备份片段的一些文件组成的集合，每个备份片段可能包含一个或多个数据库文件的备份。

注：使用此命令创建备份片段时，FORMAT 参数指定用于创建备份片段文件名时使用的模式。也可以通过 ALLOCATE CHANNEL 和 CONFIGURE 命令提供 FORMAT 规范。

创建映像副本

```
RMAN> BACKUP AS COPY DATAFILE '/ORADATA/users_01_db01.dbf';  
RMAN> BACKUP AS COPY ARCHIVELOG LIKE '/arch%';
```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

创建映像副本

映像副本是单个数据文件、归档重做日志或控制文件的克隆。可以使用 BACKUP AS COPY 命令或操作系统命令创建映像副本。使用 RMAN 的 BACKUP AS COPY 命令创建映像副本时，服务器会话将验证文件中的块，然后在控制文件中记录副本信息。

映像副本具有以下特征：

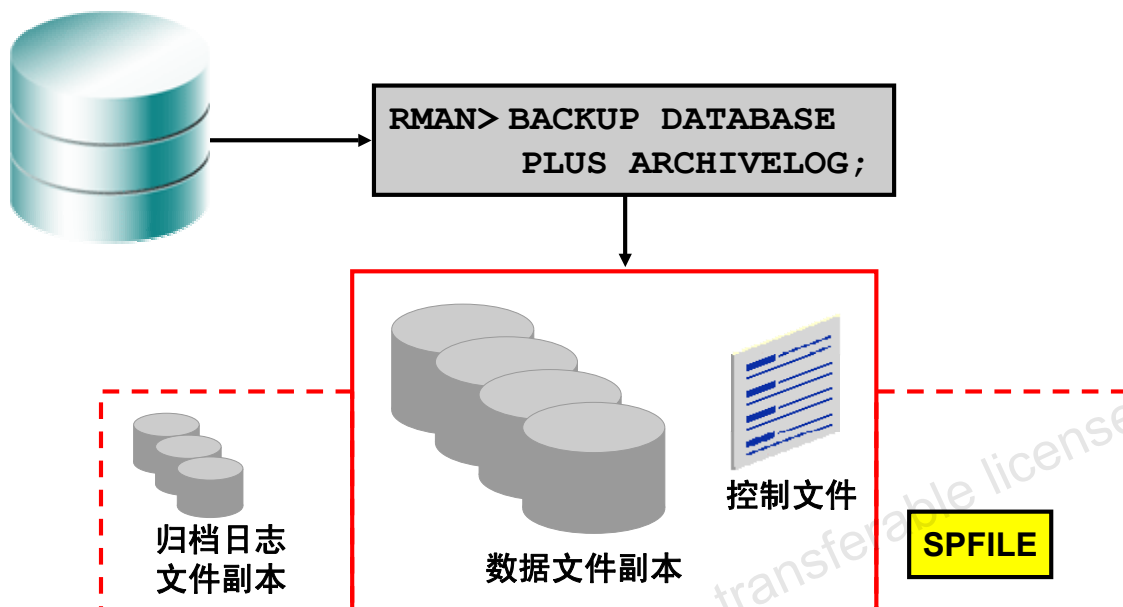
- 映像副本只能写入磁盘。处理大型文件时，复制的时间可能很长，但是由于磁盘上提供了副本，因此可明显地减少还原时间。
- 如果文件存储在磁盘上，在 RMAN 中通过 SWITCH 命令可立即使用文件，这个命令等效于 ALTER DATABASE RENAME FILE SQL 语句。
- 在映像副本中，无论块中是否包含数据都会复制所有的块，这是因为 Oracle DB 进程会复制文件，还会执行其它操作，如检查损坏的块以及在控制文件中注册副本。要加速复制过程，可使用 NOCHECKSUM 参数。默认情况下，RMAN 将计算每个备份块的校验和，并将其与备份一起存储起来。还原备份时，将对校验和进行验证。有关 BACKUP 命令的 NOCHECKSUM 选项的详细信息，请参阅《Oracle Database Backup and Recovery Reference》。
- 映像副本可以是完全备份的一部分，或是 0 级增量备份的一部分，因为文件副本中始终包含所有块。如果副本将与增量备份集一起使用，则必须使用 0 级选项。

创建映像副本（续）

幻灯片中的示例将创建以下映像副本：

- /ORADATA/users01_db01.dbf 数据文件的副本
- 归档重做日志文件的副本

创建整体数据库备份



版权所有 © 2011, Oracle。保留所有权利。

创建整体数据库备份

整体数据库备份可以是整个数据文件集的备份集或映像副本，其中必须包含控制文件。此外，还可以根据需要包括服务器参数文件 (SPFILE) 和归档重做日志文件。使用 Recovery Manager (RMAN) 可为所有数据库文件创建映像副本，这只需执行以下几步即可完成：装载或打开数据库，启动 RMAN，然后输入幻灯片中所示的 BACKUP 命令。此外，在备份归档日志文件时还可以提供 DELETE INPUT 选项。这会使 RMAN 在备份归档日志文件之后将其删除。如果未使用快速恢复区，则该选项尤为有用，它会在空间变得紧缺时删除文件，从而为您执行空间管理。如果使用该选项，则幻灯片中的命令应类似于：

```
RMAN> BACKUP DATABASE PLUS ARCHIVELOG DELETE INPUT;
```

必须在发出以下 CONFIGURE 命令后，才能按上述方式进行备份：

- CONFIGURE DEFAULT DEVICE TYPE TO disk;
- CONFIGURE DEVICE TYPE DISK BACKUP TYPE TO COPY;
- CONFIGURE CONTROLFILE AUTOBACKUP ON;

此外还可使用以下命令，为数据库中所有数据文件和控制文件的以前的映像副本创建备份（备份集或映像副本）：

```
RMAN> BACKUP COPY OF DATABASE;
```

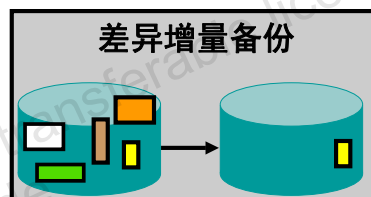
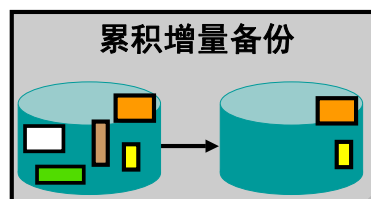
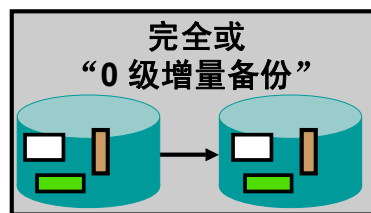
创建整体数据库备份（续）

默认情况下，RMAN 依次执行每个 BACKUP 命令。但是，可采用以下方法并行执行复制操作：

- 使用 `CONFIGURE DEVICE TYPE DISK PARALLELISM n` 命令，其中 `n` 是所需的并行度
- 分配多个通道
- 指定一个 `BACKUP AS COPY` 命令并列多个文件

RMAN 备份类型

- 完全备份包含所有使用的数据文件块。
- 0 级增量备份等同于标记为 0 级的完全备份。
- 1 级累积增量备份只包含自上次 0 级增量备份以来修改过的块。
- 1 级差异增量备份只包含自上次增量备份以来修改过的块。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

RMAN 备份类型

完全备份

完全备份不同于整体数据库备份。完全数据文件备份是包含文件中所有已用数据块的备份。RMAN 将所有块复制到备份集或映像副本中，仅跳过从未使用的数据文件块。完全映像副本可准确地再现整个文件的内容。完全备份不能成为增量备份策略的一部分；它也不能成为后续增量备份的父项。

增量备份

增量备份是 0 级备份，其中包含数据文件中除从未使用的块之外的所有块；或者是 1 级备份，其中仅包含自上次备份以来更改过的那些块。0 级增量备份在物理上与完全备份完全一样。唯一区别是 0 级备份（及映像副本）可用作 1 级备份的基础，但完全备份不可用作 1 级备份的基础。

增量备份是使用 BACKUP 命令的 INCREMENTAL 关键字指定的。可以指定 INCREMENTAL LEVEL [0 | 1]。

RMAN 备份类型（续）

RMAN 可创建多级增量备份，如下所示：

- **差异：**是增量备份的默认类型，备份自上次增量备份（1 级或 0 级）以来更改过的所有块
- **累积：**备份自上次 0 级备份以来更改过的所有块

示例

- 要执行 0 级增量备份，请使用以下命令：

```
RMAN> BACKUP INCREMENTAL LEVEL 0 DATABASE;
```
- 要执行差异增量备份，请使用以下命令：

```
RMAN> BACKUP INCREMENTAL LEVEL 1 DATABASE;
```
- 要执行累积增量备份，请使用以下命令：

```
RMAN> BACKUP INCREMENTAL LEVEL 1 CUMULATIVE DATABASE;
```

默认情况下，如果既没有指定 FULL 也没有指定 INCREMENTAL，则 RMAN 将执行完全备份。将数据文件备份到备份集时，压缩未使用的块会导致跳过从未被写过的块，甚至对于完全备份也是如此。

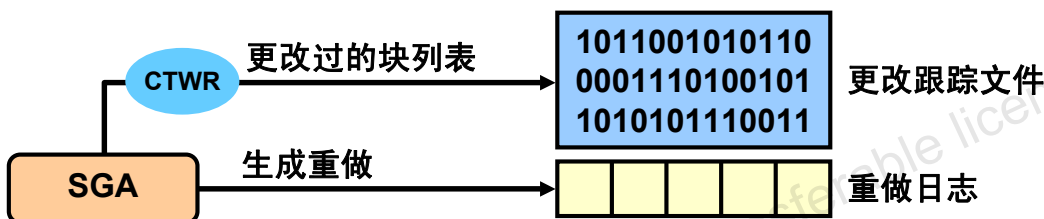
虽然完全映像副本备份可通过 RECOVER 命令应用增量备份来增量更新，但是完全备份对后续的增量备份没有影响，也不会成为任何增量备份策略的一部分。“使用 RMAN 执行恢复”一课会介绍这方面内容。

注：数据库处于 NOARCHIVELOG 模式时可执行任何类型的备份（完全或增量）— 当然，数据库必须未处于打开状态。另请注意，数据库只能恢复到上次备份时的状态。只有当数据库处于 ARCHIVELOG 模式时，才可以将其恢复到上次提交事务处理时的状态。

快速增量备份

通过块更改跟踪实现，它可以：

- 维护自上次备份以来发生更改的块的记录
- 生成重做时，将此记录写入文件
- 执行备份时自动访问，使备份更快地执行



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

快速增量备份

执行增量备份的目的是为了只备份自上一次备份以来更改过的数据块。使用 RMAN 可创建数据文件、表空间或整体数据库的增量备份。执行增量备份时，RMAN 只读取所引用的块来查找自上次备份以来发生过更改的块。这会减小备份大小，因为只备份更改过的块。此外，由于减少了需要还原的块数，因此还会加快恢复速度。

通过启用块更改跟踪，可执行快速增量备份。块更改跟踪会将有更改的每个块的物理地址写入到一个文件中。需要执行增量备份时，RMAN 可查看块更改跟踪文件，并只备份该文件所引用的块，而无需通过扫描每个块来确定该块自上次备份以来是否发生过更改。这会加快增量备份的速度。

跟踪文件的维护是完全自动进行的，不需要用户的干预。块更改跟踪文件的大小与以下各项成比例：

- 数据库大小，以字节为单位
- RAC 环境中启用的线程数
- 块更改跟踪文件维护的旧备份数

块更改跟踪文件最小为 10 MB，分配的新空间以 10 MB 为单位递增。默认情况下，Oracle DB 不记录块更改信息。

启用快速增量备份

ORACLE Enterprise Manager 10g Database Control

Database Instance: orcl > Backup Settings

Backup Settings

Device Backup Set Policy

Backup Policy

☐ Automatically backup the control file and server parameter file (SPFILE) with every backup and database structural change

Autobackup Disk Location
An existing directory or diskgroup name where the control file and server parameter file will be backed up. If you do not specify a location, the files will be backed up to the flash recovery area location.

☐ Optimize the whole database backup by skipping unchanged files such as read-only and offline datafiles that have been backed up

☐ Enable block change tracking for faster incremental backups

Block Change Tracking File
Specify a location and file, otherwise an Oracle managed file will be created in the database area.

```
ALTER DATABASE
{ENABLE|DISABLE} BLOCK CHANGE TRACKING
[USING FILE '...']
```

版权所有 © 2011, Oracle。保留所有权利。

启用快速增量备份

在 Database Control 主页上可以启用块更改跟踪功能。导航到 “Availability > Backup Settings > Policy（可用性 > 备份设置 > 策略）”。如果已设置 DB_CREATE_FILE_DEST 初始化参数，则不需要设置块更改跟踪文件的目标位置，这是因为会在 DB_CREATE_FILE_DEST 位置创建该文件，格式为 Oracle Managed File (OMF) 文件。但是，可以指定块更改跟踪文件的名称，然后将其放在选定的位置。

还可以使用 ALTER DATABASE 命令启用或禁用此功能。如果更改跟踪文件与数据库文件一起存储在数据库区域中，当禁用更改跟踪时就会删除更改跟踪文件。通过使用 ALTER DATABASE RENAME 命令可重命名块更改跟踪文件。数据库必须处于 MOUNT 状态才能重命名跟踪文件。使用 ALTER DATABASE RENAME FILE 命令更新控制文件时可引用新位置。可以使用以下语法更改块更改跟踪文件的位置：

```
ALTER DATABASE RENAME FILE '...' TO '...';
```

注：RMAN 不支持对块更改跟踪文件进行备份和恢复。因此，不应将其放在快速恢复区。

监视块更改跟踪

```
SQL> SELECT filename, status, bytes
2 FROM v$dbblock_change_tracking;
```

```
SQL> SELECT file#, avg(datafile_blocks),
2         avg(blocks_read),
3         avg(blocks_read/datafile_blocks)
4         * 100 AS PCT_READ_FOR_BACKUP,
5         avg(blocks)
5 FROM v$backup_datafile
6 WHERE used_change_tracking = 'YES'
7 AND incremental_level > 0
8 GROUP BY file#;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

监视块更改跟踪

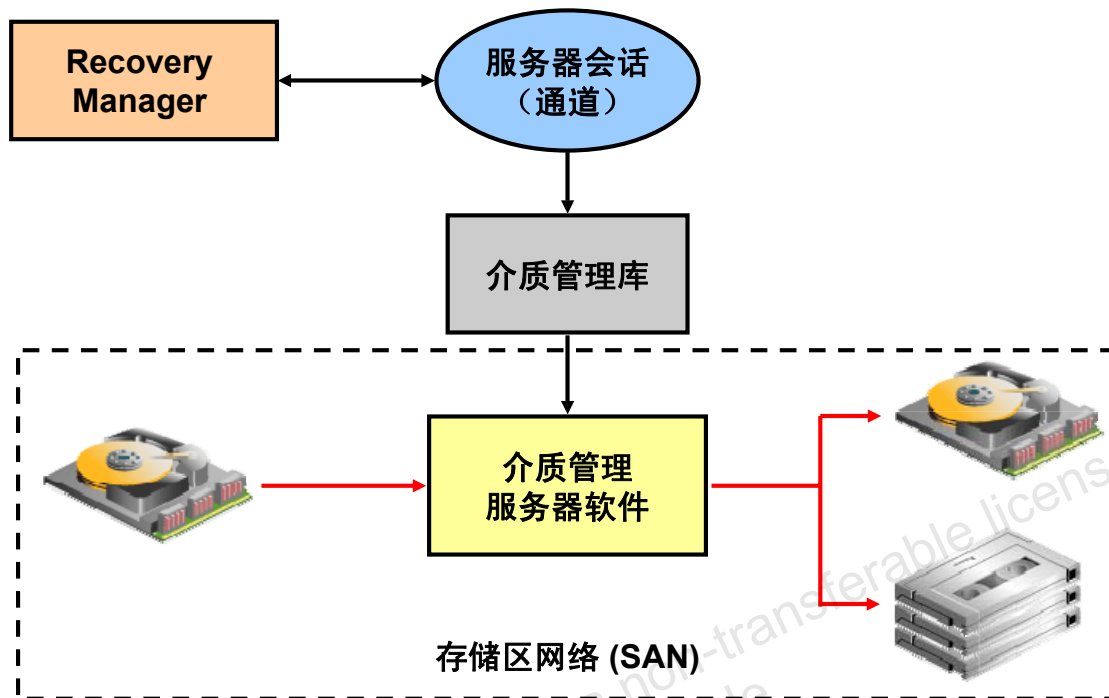
在 V\$BLOCK_CHANGE_TRACKING 视图的输出中会显示块更改跟踪文件的位置、块更改跟踪的状态 (ENABLED/DISABLED) 和文件大小（以字节为单位）。

对 V\$BACKUP_DATAFILE 视图进行查询，可显示块更改跟踪功能对最大限度减少增量备份 I/O 的作用（PCT_READ_FOR_BACKUP 列）。值比较高时表示 RMAN 在增量备份期间从数据文件中读取的块非常多。通过减少增量备份之间的时间间隔可降低这个比率。

以下是 V\$BACKUP_DATAFILE 查询的输出格式示例：

FILE#	BLOCKS_IN_FILE	BLOCKS_READ	PCT_READ_FOR_BACKUP	BLOCKS_BACKED_UP
1	56320	4480	7	462
2	3840	2688	70	2408
3	49920	16768	33	4457
4	640	64	10	1
5	19200	256	1	91

执行代理副本



版权所有 © 2011, Oracle。保留所有权利。

执行代理副本

使用 RMAN BACKUP 命令的 PROXY 选项可以请求 MML 执行文件的副本。

语法：

```
BACKUP [AS BACKUPSET] ... PROXY [ONLY] DATABASE|TABLESPACE....
```

PROXY ONLY 选项对于备份由代理完成的介质管理器和存储网络非常有用，这可以明显地减少了存储网络通信量。

一些介质管理产品可完善地管理 Oracle 数据文件与备份设备之间的所有数据移动。某些产品在存储子系统与介质子系统之间使用了高速连接，从而可大幅减少主数据库服务器的备份负载。这一点十分有用，因为复制是在 SAN 上而不是在 LAN 上执行的。除了通过 LAN 在 MML 之间传送状态之外，RMAN 此时已不再涉及操作。

使用 BACKUP COPIES 创建双向备份集

```
RMAN> BACKUP AS BACKUPSET DEVICE TYPE sbt  
2> COPIES 2  
3> INCREMENTAL LEVEL 0  
4> DATABASE;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 BACKUP COPIES 创建双向备份集

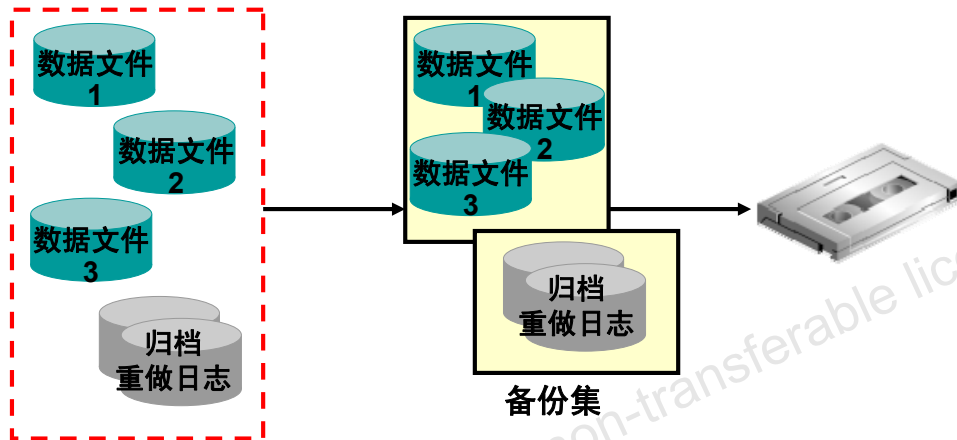
为了创建双向备份集，可使用带有 COPIES 选项的 BACKUP 命令覆盖其它 COPIES 或 DUPLEX 设置。

要使用 BACKUP COPIES 创建双向备份，请执行以下步骤：

1. 使用 BACKUP 命令的 COPIES 选项指定完全一样的副本的数量。
2. 发出 LIST BACKUP 命令来验证备份。

创建备份集的备份

```
RMAN> BACKUP DEVICE TYPE DISK AS BACKUPSET  
2> DATABASE PLUS ARCHIVELOG;  
RMAN> BACKUP DEVICE TYPE sbt BACKUPSET ALL;
```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

创建备份集的备份

使用 **RMAN BACKUP BACKUPSET** 命令可备份先前创建的备份集。使用 **RMAN** 只能备份在设备类型 **DISK** 上创建的备份集。备份集可备份到任何可用的设备类型。

BACKUP BACKUPSET 命令使用默认磁盘通道在磁盘之间复制备份集。要从磁盘备份到磁带，必须配置或手动分配非磁盘通道。

备份只读表空间

备份只读表空间的注意事项：

- 仅当符合保留策略的备份不存在时，备份优化才会让 RMAN 备份只读表空间。
- 如果将表空间更改为可读/写，请直接对其进行备份。
- 可使用 RMAN BACKUP 命令的 SKIP READONLY 选项跳过只读表空间或数据文件。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

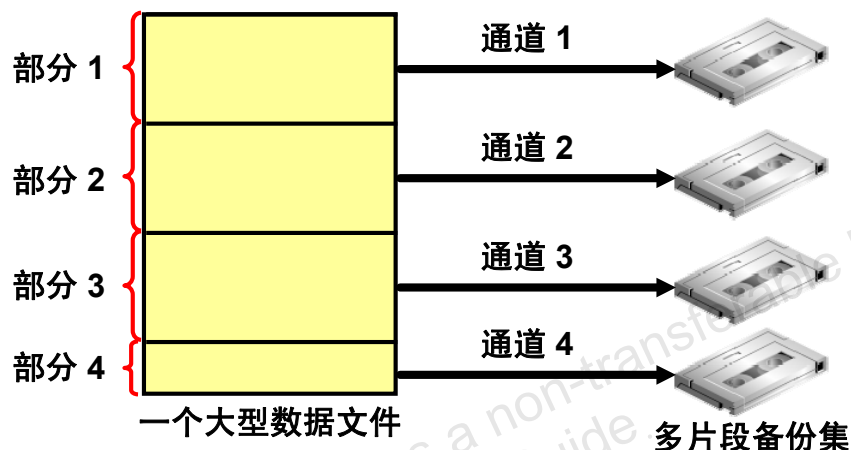
备份只读表空间

由于只读表空间不能写入，因此无需像读/写表空间那样进行连续备份。可以使用 BACKUP 命令的 SKIP READONLY 选项指示 RMAN 不备份只读表空间。

为超大型文件配置备份和还原

单个文件的多部分备份：

- 由 RMAN 创建，具有指定的大小值
- 独立进行处理（串行或并行）
- 生成多片段备份集
- 提高了备份的性能



版权所有 © 2011, Oracle。保留所有权利。

为超大型文件配置备份和还原

Oracle 数据文件最大为 128 TB。通常，RMAN 备份的最小单元是整个文件。这对于如此大的文件不适用。RMAN 可以选择将大型文件拆分为多个部分，然后单独对这些部分进行备份和还原。这是通过创建多部分备份实现的，这种备份将为备份集生成的文件拆分为多个独立文件。这种备份仅适用于备份集，不适用于映像副本。

每个文件部分都是文件中一系列连续的块。可按串行或并行方式独立处理每个文件部分。将文件备份到各单独部分可以提高备份操作的性能，还允许重新启动大型文件备份。

多部分备份作业可生成一个多片段备份集。每个片段都包含一个文件部分。多部分备份的所有部分的大小都相同（最后一部分除外）。每个文件最多有 256 个部分。

注：对驻留在少量磁盘上的大型文件进行备份时，不宜使用很高的并行度，因为这会使并行操作的作用失效；同时对同一磁盘设备进行多个访问会导致争用。

此功能已内置到 RMAN 中。除 Oracle Database 11g 的常规安装外，不需要其它安装。因为早期版本无法还原多部分备份，所以必须至少将 COMPATIBLE 设置为 11.0。

创建 RMAN 多部分备份

RMAN 命令语法:

```
BACKUP <options> SECTION SIZE <integer> [K | M | G]
```

```
VALIDATE DATAFILE <options> SECTION SIZE <integer> [K | M | G]
```

示例:

```
RMAN> BACKUP DATAFILE 5 SECTION SIZE = 25M TAG 'section25mb';
backing up blocks 1 through 3200
piece handle=/u01/.../o1_mf_nnndf_SECTION25MB_382dryt4_.bkp
tag=SECTION25MB comment=NONE
...
backing up blocks 9601 through 12800
piece handle=/u01/.../o1_mf_nnndf_SECTION25MB_382dsto8_.bkp
tag=SECTION25MB comment=NONE
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

创建 RMAN 多部分备份

BACKUP 和 VALIDATE DATAFILE 命令接受以下选项:

```
SECTION SIZE <integer> [K | M | G]
```

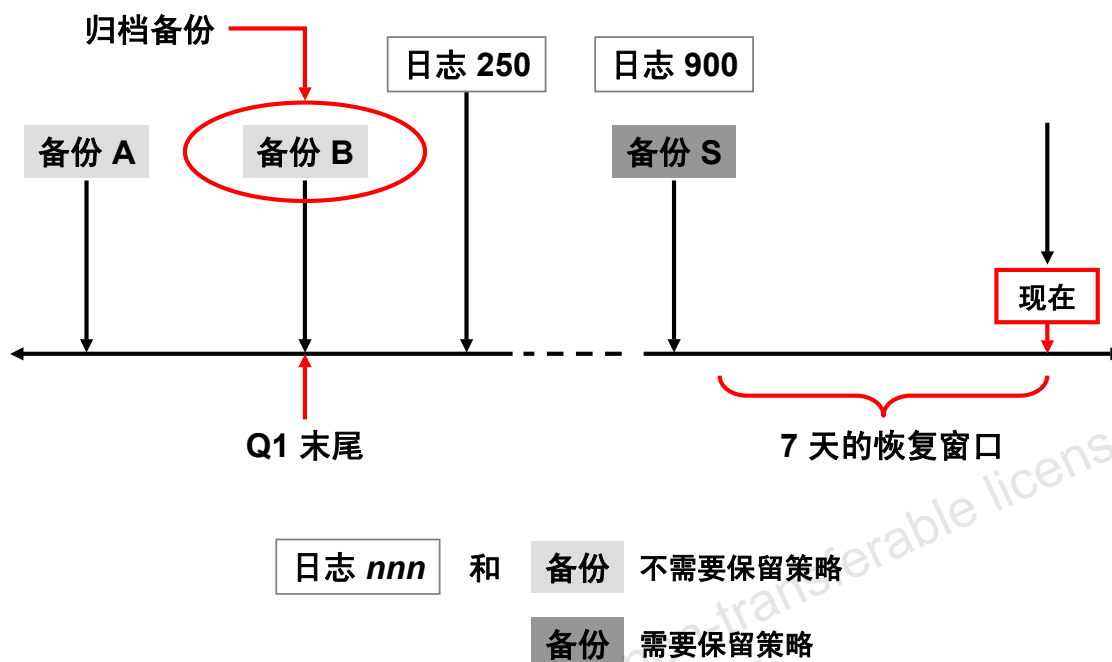
使用该选项可以为每个备份部分指定计划大小。该选项既是备份命令级别选项也是备份规范级别选项, 所以可在同一备份作业中对不同的文件应用不同的部分大小。

在幻灯片的示例中, 将对数据文件 5 进行备份, 并将部分大小指定为 25 MB。该数据文件大小为 100 MB, 因此将创建四个部分。请注意, 如块范围所示, 将块写入部分文件时会保持块相邻性。

查看有关特定多部分备份的元数据

- V\$BACKUP_SET 和 RC_BACKUP_SET 视图都有一个 MULTI_SECTION 列, 用于表明是否为多部分备份。
- V\$BACKUP_DATAFILE 和 RC_BACKUP_DATAFILE 视图都有一个 SECTION_SIZE 列, 用于指定多部分备份的每个部分中的块数。零意味着对整个文件进行备份。

归档备份：概念



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

归档备份：概念

如果需要在指定时间内保留联机备份，RMAN 通常会假定您可能需要在自执行该备份以来到现在之间的任意时间执行时间点恢复。为了满足这一要求，RMAN 会在此时段内保留归档日志。但是，您可能仅需要在指定的时间（如两年）内保留特定备份（并使其保持一致和可恢复）。您不打算恢复到自执行该备份以后的某一时间点，只是希望能够正好恢复到执行该备份的确切时间。此外，您还需要维护保留策略以使备份井然有序，因此无法使备份恢复到两年前。为了满足保留数据的商业或法律要求，通常需要这么做。

归档备份可以解决这一问题。如果将某一备份标记为归档备份，该属性将覆盖为此备份目的配置的所有保留策略。保留归档备份时，可将其指定为仅在某一特定时间过时，也可以将其指定为永不过时。如果要指定后者，则需要使用恢复目录。

KEEP 子句会创建一个归档备份，此备份是某个时间点的数据库快照。仅保留将此备份还原至一致状态所需的重做日志。在备份完成后发出的 RESTORE POINT 子句确定保留的重做日志数（足以将备份还原到 RESTORE POINT 时间）。

归档备份还可保证包含还原备份所需的全部文件。RMAN 包含数据文件、SPFILE、归档日志文件（仅限恢复联机备份所需的那些文件）及相关自动备份文件。所有这些文件都必须保存到同一介质系列（或磁带组）中。

归档备份：概念（续）

此外，还可指定要创建的还原点，该还原点与归档备份具有相同的 SCN。实际上，这为执行备份的时间点提供了一个有意义的名称。

创建归档备份之后，它将保留指定的时间。即使具有非常短的保留窗口并运行了 `DELETE OBSOLETE` 命令，归档备份也会保留下来。

此备份是数据库在某个时间点的快照，可用于将数据库还原到另一个主机（例如，用于测试目的）。

注：归档备份不能写入到快速恢复区。因此，如果具有快速恢复区，则必须通过 `FORMAT` 子句指定其它位置。

使用 EM 创建归档备份

Schedule Customized Backup: Settings

Database **orcl.us.oracle.com** Cancel Back Step 3 of 5 Next

Backup Strategy **Customized Backup**

Object Type **Tablespaces**

These are the settings for your current backup job. You can select your backup destination directly from this page. You can also view the default settings or override the settings by clicking the buttons below.

☒ Disk

Disk Backup Location **/archive_bu**

☐ Tape

Media Management Vendor(MMV) Library Parameters **not specified**

View Default Settings Override Current Settings

Changed settings will only apply to the current

Override Retention Policy

☐ Do not override the retention policy

☒ Keep this backup for the specified number of days Days **365**

Device Backup Set Policy

Cancel OK

版权所有 © 2011, Oracle。保留所有权利。

使用 EM 创建归档备份

要使用 Oracle Enterprise Manager 创建归档备份，请执行以下步骤：

1. 选择“Availability > Schedule Backup > Schedule Customized Backup（可用性 > 调度备份 > 调度定制备份）”。
2. 按照“Schedule Customized Backup（调度定制备份）”向导中的步骤进行操作，直到进入“Settings（设置）”页。
3. 单击“Override Current Settings（覆盖当前设置）”，然后单击“Policy（策略）”选项卡。在“Override Retention Policy（覆盖保留策略）”部分中，可选择将备份保留指定的天数。将根据备份作业名称生成还原点。此外，还可能需要在备份文件指定其它目标位置；为此，应使用“Device（设备）”选项卡。

使用 KEEP 选项创建的备份包括 SPFILE、控制文件、还原此备份所需的归档重做日志文件以及数据文件。此备份是数据库在某个时间点的快照，可用于将数据库还原到另一个主机。

使用 RMAN 创建归档备份

- 指定 KEEP 子句，当数据库联机时包括数据文件和归档日志备份集：

```
KEEP {FOREVER | UNTIL TIME [=] ' date_string '}
NOKEEP
[RESTORE POINT rsname]
```

- 列出 RMAN 资料档案库已知的所有还原点：

```
LIST RESTORE POINT ALL;
```

- 显示特定的还原点：

```
LIST RESTORE POINT 'rsname';
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 RMAN 创建归档备份

可使用 RMAN 通过以下语法创建归档备份：

```
BACKUP ... KEEP {FOREVER|UNTIL TIME 'SYSDATE + <n>'} RESTORE POINT
<restore_point_name>
```

使用 UNTIL TIME 子句可指定保留策略对归档备份失效的时间。也可指定 FOREVER，这意味着备份始终是归档备份，除非执行其它操作进行更改。

此外，还可使用 RESTORE POINT 子句指定要与此备份关联的还原点的名称。RESTORE POINT 子句将在控制文件中创建“一致性”点。该子句为特定的 SCN 分配一个名称。SCN 是在完成备份数据文件后捕获的。可将归档备份还原并恢复到此时间点，使数据库处于打开状态。与之相反，UNTIL TIME 子句指定了备份必须保留到的日期。

管理归档数据库备份

① 归档数据库备份:

```
RMAN> CONNECT TARGET /  
RMAN> CONNECT CATALOG rman/rman@catdb  
RMAN> CHANGE BACKUP TAG 'consistent_db_bkup'  
2> KEEP FOREVER;
```

② 更改数据库副本的状态:

```
RMAN> CHANGE COPY OF DATABASE CONTROLFILE NOKEEP;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理归档数据库备份

CHANGE 命令会更改与配置的保留策略有关的备份或副本的排除状态。例如，可以指定 CHANGE ... NOKEEP 使当前排除在保留策略之外的备份变为 OBSOLETE 状态。

第一个示例将一致的备份更改为归档备份，您计划在场外存储该归档备份。因为该数据库是一致的，所以无需对其进行恢复，不需要与备份一起保存归档重做日志。

第二个示例指定数据文件和控制文件的任何长期映像副本都不应处于排除状态，而应根据现有保留策略变为过期。此语句实际上会将归档属性从这些备份文件中删除。如果未指定标记（如此示例所示），则 CHANGE 执行会应用于指定类型的所有备份。应指定一个标记，以便只更改需要更改的备份文件。

注：RESTORE POINT 选项不适用于 CHANGE 命令，因为无法为已过去的时间（创建备份的时间）创建还原点。

备份恢复文件

- 只备份快速恢复区中的文件：

```
RMAN> BACKUP RECOVERY AREA
```

- 备份所有恢复文件：

```
RMAN> BACKUP RECOVERY FILES
```



快速恢复区



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

备份恢复文件

有两种方法可备份恢复数据。BACKUP RECOVERY AREA 命令可备份当前或以前任何快速恢复区中找到的所有文件。BACKUP RECOVERY FILES 命令可备份所有恢复文件，即使这些文件不在 FRA 中。使用后一种方法可以更好地保护数据，使其免于丢失，例如，它将备份不在快速恢复区中的控制文件或数据文件的任何副本。

默认情况下，备份优化对这两个命令有效，即使使用 CONFIGURE 命令禁用了该功能也是如此。这意味着此命令仅备份尚未备份的恢复文件。可以使用 FORCE 选项来强制备份所有文件。

不能为这两个命令中的任何一个指定 DEVICE TYPE DISK。

注：RMAN 仅备份数据库文件：数据文件、控制文件、SPFILE、归档日志文件和这些文件的备份。将某一操作系统文件置于快速恢复区将导致恢复区备份中包含该操作系统文件。

管理备份：报告

使用以下 RMAN 命令可获取有关备份的信息：

- LIST：显示资料档案库中记录的有关备份集、代理副本和映像副本的信息
- REPORT：生成资料档案库的详细分析信息
- REPORT NEED BACKUP：列出需要备份的所有数据文件
- REPORT OBSOLETE：标识按照备份保留策略不再需要的文件

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理备份：报告

使用 RMAN LIST 命令可显示资料档案库中记录的有关备份集、代理副本和映像副本的信息。使用这个命令可列出：

- RMAN 资料档案库中不具有 AVAILABLE 状态的备份和副本
- 可用的且可以用于还原操作的数据文件备份和副本
- 备份集和副本，其中包含指定数据文件列表或指定表空间的备份
- 包含指定名称或范围的所有归档日志备份的备份集和副本
- 由标记、完成时间、可恢复性或设备限制的备份集和副本
- 指定数据库或资料档案库已知的所有数据库的原型
- 恢复目录中的存储脚本

使用 RMAN REPORT 命令可更详细地分析 RMAN 资料档案库中的信息。

REPORT NEED BACKUP 命令用于标识需要备份的所有数据文件。报告假定还原时会使用最近的备份。

管理备份：报告（续）

使用 `REPORT OBSOLETE` 命令可标识按照备份保留策略不再需要的文件。默认情况下，`REPORT OBSOLETE` 命令报告在当前配置的保留策略下已过时的文件。使用 `REPORT OBSOLETE` 命令的 `REDUNDANCY` 或 `RECOVERY WINDOW` 保留策略选项，可根据不同的保留策略生成过时文件的报告。

有关详细语法的信息，请参阅《Oracle Database Backup and Recovery Reference》。

管理备份：动态性能视图

在目标数据库中查询以下动态性能视图可获得有关备份的信息：

- V\$BACKUP_SET：创建的备份集
- V\$BACKUP_PIECE：存在的备份片段
- V\$DATAFILE_COPY：磁盘上的数据文件副本
- V\$BACKUP_FILES：有关创建备份时所创建的所有文件的信息

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理备份：动态性能视图

有许多视图都提供了与备份有关的信息。幻灯片中显示的是最常用的视图。

如果使用的是恢复目录，可查询相应的视图，对于在恢复目录数据库中注册的每个目标数据库，这些视图都包含相同的信息。这些相应的视图具有相同的名称，只是“V\$”被替换为“RC_”。另外，这些视图还位于恢复目录所有者拥有的方案中。例如，恢复目录中显示幻灯片中所示信息的相应视图包括：RC_BACKUP_SET、RC_BACKUP_PIECE、RC_DATAFILE_COPY 和 RC_BACKUP_FILES。

要查询 RC_BACKUP_FILES 视图，必须先在恢复目录数据库中执行以下语句：

```
SQL> CALL DBMS_RCVMAN.SETDATABASE(null,null,null,<dbid>);
```

其中 <dbid> 是目标数据库的数据库 ID。

使用 Oracle Enterprise Manager 查看备份报告

Results		
Input Summary		
Datafile	Control File	SPFile
Files Backed Up 15	Files Backed Up 3	Files Backed Up 3
Distinct Files 5	Distinct Files 1	Distinct Files 2
Distinct Tablespaces 5	Total Size 28.92M	Total Size 0.01K
Total Size 5.26G	Oldest Checkpoint Time Aug 14, 2007 8:04:16 PM	Oldest Modification Time Aug 14, 2007 7:54:20 PM
Oldest Checkpoint Time Aug 14, 2007 9:19:52 PM	Newest Checkpoint Time Aug 14, 2007 9:21:59 PM	Newest Modification Time Aug 14, 2007 9:01:45 PM
Newest Checkpoint Time Aug 14, 2007 9:21:52 PM		

View Backup Report

The following backup jobs are known to the database. The data is retrieved from the database control file.

Search

Status Start Time Type

Results

Total **19** (Completed **15** Failed **4**)

Backup Name	Status	Start Time	Time Taken	Type	Output Devices	Input Size	Output Size	Output Rate (Per Sec)
2007-08-15T00:18:04	COMPLETED	Aug 15, 2007 12:18:10 AM GMT+07:00	00:01:12	DATAFILE FULL	DISK	1.14M	304.00K	4.22K
2007-08-14T19:55:28	COMPLETED	Aug 14, 2007 7:56:20 PM GMT+07:00	01:25:50	DB INCR	DISK	1.86G	1.80G	367.41K
2007-08-14T02:52:20	COMPLETED	Aug 14, 2007 2:52:22 AM GMT+07:00	00:00:09	CONTROLFILE	SBT_TAPE	9.63M	0.00K	0.00K

Manage

[Schedule Backup](#)
[Manage Current Backups](#)
[Backup Reports](#)
[Manage Restore Points](#)
[Perform Recovery](#)
[View and Manage Transactions](#)

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

使用 Oracle Enterprise Manager 查看备份报告

在“Backup Report（备份报告）”页上，可显示数据库通过数据库控制文件中记录的备份作业信息获得的备份作业列表。

使用页顶部的“Search（搜索）”字段，可自定义显示在“Result（结果）”表中的作业。

“Results（结果）”表中列出了每个备份作业的基本信息，如备份作业的“Start Time（开始时间）”、“Time Taken（所用时间）”和“Status（状态）”。还可通过“Backup Name（备份名称）”列中的链接，使用“Results（结果）”表来查看各个备份作业的详细报告。

通过单击“Results（结果）”表中作业的“Status（状态）”，可显示备份作业的“Summary of job（作业概要）”页，可在其中查看输出日志的内容。

单击“Backup Name（备份名称）”链接，然后可以使用“Backup Report（备份报告）”页显示有关该备份的详细信息。此页显示的信息来自数据库控制文件中记录的信息。

“Backup Report（备份报告）”页在“Result（结果）”部分显示各类结果信息，如包含有关已备份文件的汇总信息的“Input Summary（输入概要）”，包含有关备份集和映像副本的汇总信息的“Output Summary（输出概要）”，然后是“Inputs and Outputs（输入和输出）”部分，其中显示包含有关数据文件、控制文件、备份集、备份片段和映像副本的作业详细资料的表。

管理备份：交叉检查和删除

使用以下 RMAN 命令可管理备份：

- **CROSSCHECK**：针对介质（如磁盘或磁带）验证在 RMAN 资料档案库中记录的备份和副本的状态
- **DELETE EXPIRED**：只删除资料档案库中状态为 EXPIRED 的文件
- **DELETE OBSOLETE**：删除不再需要的备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

管理备份：交叉检查和删除

使用 **CROSSCHECK** 命令可确保恢复目录或控制文件中备份的相关数据与磁盘或介质管理目录中的实际文件同步。**CROSSCHECK** 命令只对 RMAN 资料档案库中记录的文件有效。

CROSSCHECK 命令检查 DISK 通道中磁盘上的文件或查询 sbt 通道中的介质管理器，但只检查标记为 AVAILABLE 或 EXPIRED 的对象。**CROSSCHECK** 命令将无法找到的所有文件的资料档案库记录更新为 EXPIRED。此命令并不删除无法找到的文件。

DELETE 命令可删除所有可应用 **LIST** 和 **CROSSCHECK** 命令的文件。例如，可以删除备份集、归档重做日志和数据文件副本。**DELETE** 命令还可同时删除物理文件及该文件的目录记录。**DELETE OBSOLETE** 命令删除不再需要的备份。它与 **REPORT OBSOLETE** 使用相同的 **REDUNDANCY** 和 **RECOVERY WINDOW** 选项。

如果不使用 **RMAN** 删除备份，则可使用 **UNCATALOG** 命令从恢复目录中删除文件，或可使用 **CROSSCHECK** 和 **DELETE EXPIRED** 命令。

有关详细语法的信息，请参阅《Oracle Database Backup and Recovery Reference》。

测验

完全数据库备份可用作增量备份的基础。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2

测验

数据库关闭时，RMAN 始终可以执行备份。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1

小结

在本课中，您应该已经学会：

- 创建映像文件备份
- 创建整体数据库备份
- 创建完全数据库备份
- 启用快速增量备份
- 创建双向备份集
- 备份备份集
- 创建 RMAN 多部分备份
- 创建归档备份以用于长期保留
- 报告和维护备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 5 概览：创建备份

本练习包含以下主题：

- 执行归档备份
- 启用块更改跟踪
- 从损坏块恢复
- 报告现有备份
- 备份控制文件

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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

6 还原和恢复任务

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

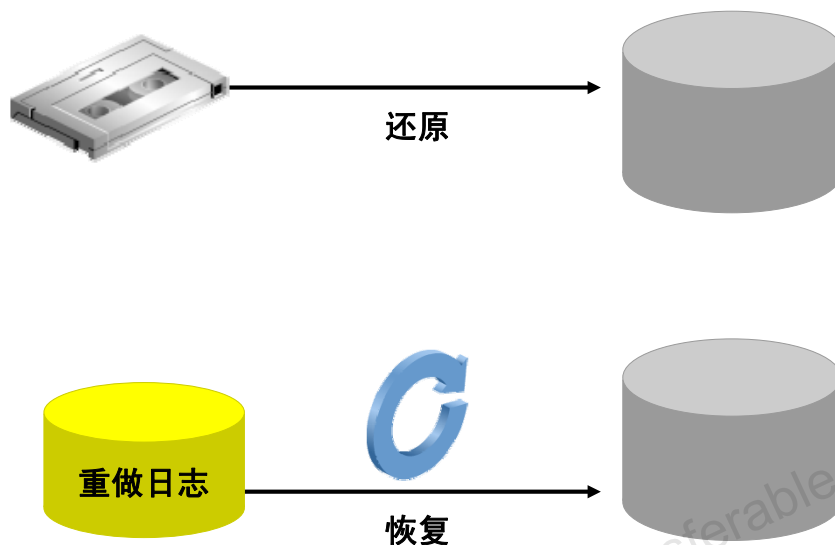
学完本课后，应能完成以下工作：

- 描述文件丢失的原因并确定相应的操作
- 描述主要的恢复操作
- 备份和恢复控制文件
- 在丢失了重做日志组后进行恢复

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

还原和恢复



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

还原和恢复

备份和恢复任务的“恢复”部分包括的活动有两种主要类型：还原和恢复。“还原”文件是指将备份复制到位以供数据库使用的过程。例如，如果由于文件所在的物理磁盘出现故障而使文件受损，则还原文件是必要的。出现这种情况通常是由于硬件问题，如磁盘写入错误或控制器故障。在这种情况下，需要将文件备份复制到新的（或已修复的）磁盘上。

“恢复”文件需要应用重做，以便将文件的状态恢复到所需的任何时间点。该点通常应尽可能地接近故障时间。

在数据库行业中，这两个操作通常全部用一个术语“恢复”来指代。

文件丢失的原因

以下原因可能会导致文件丢失：

- 用户错误
- 应用程序错误
- 介质故障



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

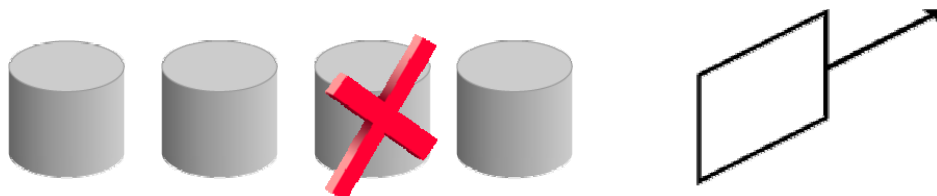
文件丢失的原因

以下原因可能会导致文件丢失或损坏：

- **用户错误：**管理员可能因疏忽而删除或覆盖了必需的操作系统文件。
- **应用程序错误：**应用程序或脚本也可能包含逻辑错误，当它处理数据库文件时，会导致文件丢失或损坏。
- **介质故障：**磁盘驱动器或控制器可能会完全或部分发生故障，从而导致文件损坏，甚至文件完全丢失。

关键性与非关键性

非关键性文件丢失是指数据库可以继续运行的故障。



通过执行以下操作之一，可以解决该问题：

- 创建一个新文件。
- 重建文件。
- 恢复丢失或损坏的文件。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

关键性与非关键性

非关键性文件是指数据库和大多数应用程序没有它也能继续运行的文件。例如，如果数据库丢失了一个多路复用重做日志文件，仍可使用其它重做日志文件副本来保持数据库持续运行。

虽然丢失非关键性文件不会导致数据库崩溃，但它会削弱数据库的功能。例如：

- 丢失索引表空间会导致应用程序和查询的运行速度大幅减慢，或者，如果这些索引用于强制实施约束，则丢失后甚至会导致应用程序无法使用。
- 只要丢失的联机重做日志组不是当前的联机日志组，就可能会导致数据库操作挂起（当 LGWR 下次尝试写入组时），直到生成新的日志文件为止。
- 丢失临时表空间会使用户无法运行查询或创建索引，直到将这些用户分配到新的临时表空间为止。

自动恢复临时文件

如果缺失任何一个临时文件，则需要临时空间来执行的 SQL 语句会失败。

```
SQL> select * from big_table order by
1,2,3,4,5,6,7,8,9,10,11,12,13;
select * from big_table order by
1,2,3,4,5,6,7,8,9,10,11,12,13
      *
ERROR at line 1:
ORA-01565: error in identifying file
'/u01/app/oracle/oradata/orcl/temp01.dbf'
ORA-27037: unable to obtain file status
Linux Error: 2: No such file or directory
```

好消息：

- 启动时自动重新创建临时文件
- （也可以手动重新创建）

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

自动恢复临时文件

如果属于临时表空间的某个临时文件 (tempfile) 丢失或损坏，则该文件中的区将不可用。在执行需要临时空间进行排序的 SQL 语句期间，此问题会声明自己为错误。

幻灯片中显示的 SQL 语句有很长一串要作为排序依据的列，因此需要临时空间。执行需要排序的此语句时会遇到缺失文件错误。

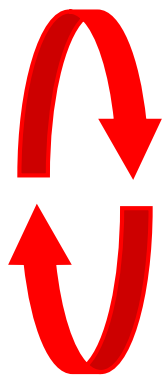
Oracle DB 实例可以在临时文件缺失的情况下启动。启动数据库实例时如果有任何临时文件不存在，系统都会自动创建这些临时文件，从而使数据库可以正常打开。发生这种情况时，启动过程中会在预警日志中显示类似下面的消息：

```
Re-creating tempfile /u01/app/oracle/oradata/orcl/temp01.dbf
```

如果您认为手动重新创建能更好地为您服务（这种情况的可能性不大），请使用下列命令：

```
SQL> ALTER TABLESPACE temp ADD TEMPFILE
'/u01/app/oracle/oradata/orcl/temp02.dbf' SIZE 20M;
SQL> ALTER TABLESPACE temp DROP TEMPFILE
'/u01/app/oracle/oradata/orcl/temp01.dbf';
```


日志组状态：综述



在任何给定时间，重做日志组的状态都会是以下值之一：

- **CURRENT**: LGWR 进程当前正在向该重做日志组写入重做数据。
- **ACTIVE**: 不再向该重做日志组写入数据，但是恢复实例时仍然需要它。
- **INACTIVE**: 不再向该重做日志组写入数据，且恢复实例时也不再需要它。

ORACLE

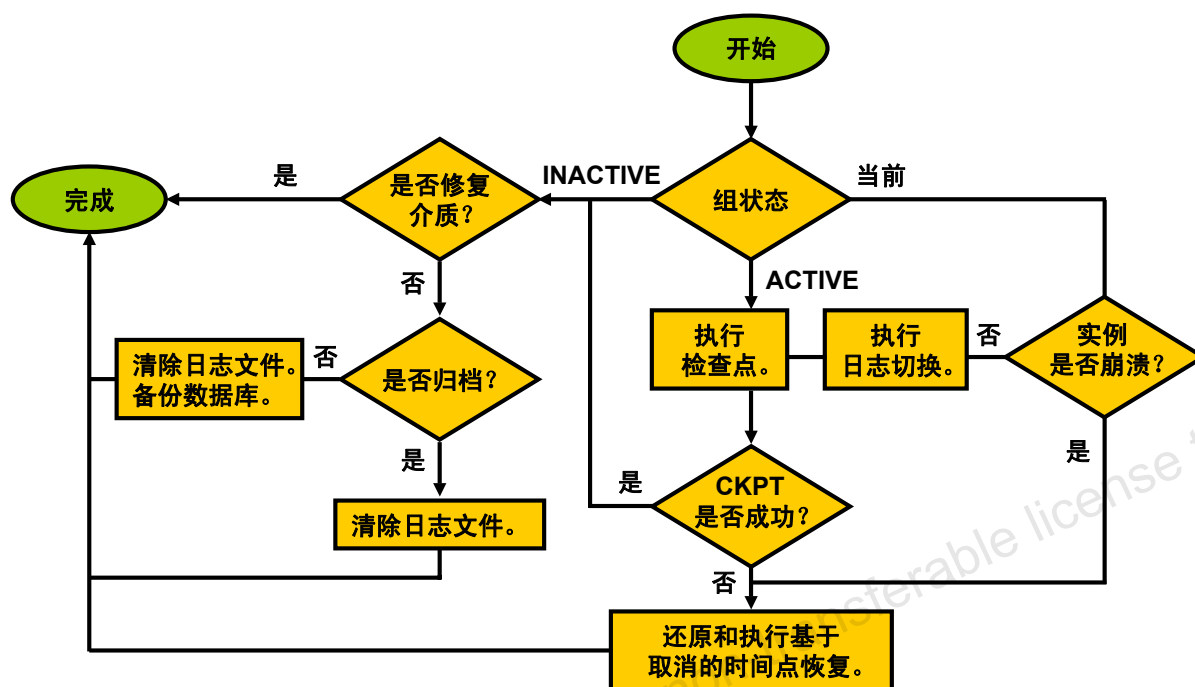
版权所有 © 2011, Oracle。保留所有权利。

日志组状态：综述

要处理重做日志文件的丢失问题，了解重做日志组的可能状态非常重要。在 Oracle DB 正常运行过程中，重做日志组会循环经历三种不同的状态。按照循环的顺序，状态分别是：

- **CURRENT**: 此状态表明 LGWR 正在向该重做日志组写入数据，以记录数据库中正在进行的所有事务处理的重做数据。该日志组将保持此状态，直到切换至其它日志组为止。
- **ACTIVE**: 该重做日志组仍包含恢复实例所需的重做数据。这是尚未执行检查点时重做日志组所处的状态，检查点会将重做日志组中出现的所有数据更改写出到数据文件。
- **INACTIVE**: 已实际执行了上面讨论的检查点，这意味着恢复实例时不再需要该重做日志组，它可以变为下一个 CURRENT 日志组。

在丢失了重做日志组后进行恢复



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

在丢失了重做日志组后进行恢复

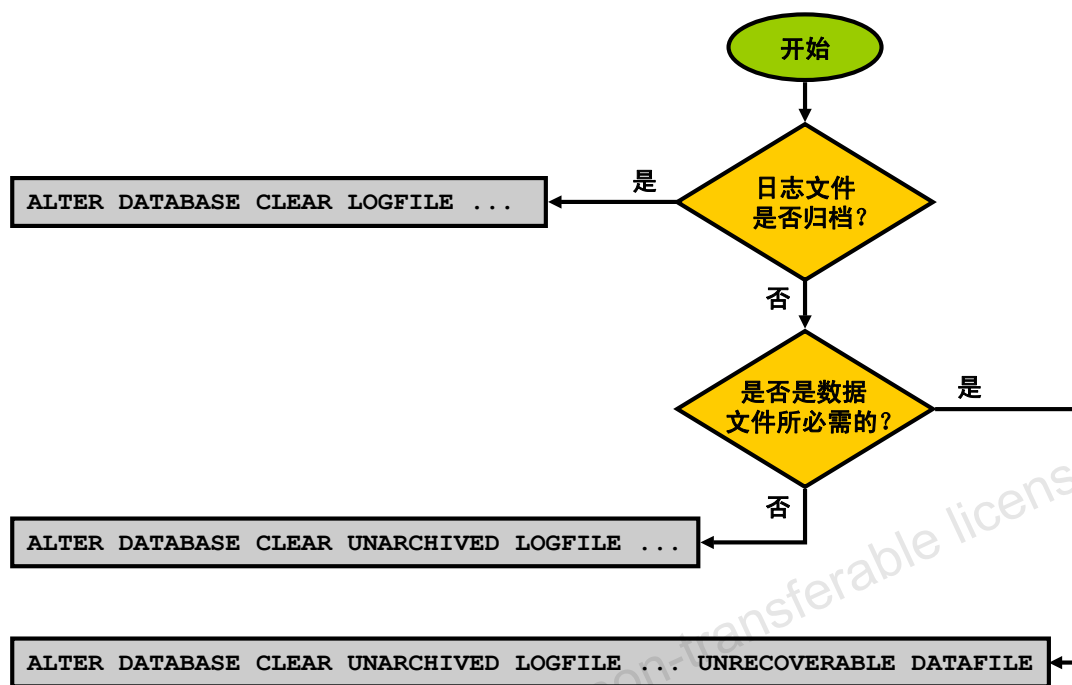
如果丢失了整个重做日志组，则该组的所有日志文件副本都是无法使用的或已丢失。

最简单的情况就是重做日志组处于 INACTIVE 状态。这意味着，当前没有向重做日志组写入数据，且不再需要它来恢复实例。如果该问题是临时的，或者您可以修复介质，则数据库将继续正常运行，在发生足够的日志切换事件后将重用该组。否则，如果无法修复介质，则可以清除日志文件。清除日志文件即表明可以重用该文件。

如果有问题的重做日志组处于 ACTIVE 状态，则即使当前没有向该重做日志组写入数据，执行实例恢复时也仍然需要它。如果可以执行检查点，则不再需要该日志文件组来恢复实例，您可以继续操作，就好像该组处于非活动状态一样。

如果该日志组处于 CURRENT 状态下，则表示在该日志组丢失时，正在向其中写入数据。在这种情况下，您甚至可能看到 LGWR 进程失败。如果发生这种情况，则实例将崩溃。此时的唯一选择就是从备份还原，执行基于取消的时间点恢复，然后使用 RESETLOGS 选项打开数据库。

清除日志文件



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

清除日志文件

可使用以下命令清除日志文件：

```
ALTER DATABASE CLEAR [UNARCHIVED] LOGFILE GROUP <n>
[UNRECOVERABLE DATAFILE]
```

清除日志文件即表明可以重用该文件。如果日志文件已归档，则可以使用该命令的最简单的形式。使用以下查询可确定哪些日志组已归档：

```
SQL> SELECT GROUP#, STATUS, ARCHIVED FROM V$LOG;
```

例如，以下命令可清除已归档的重做日志组 3：

```
SQL> ALTER DATABASE CLEAR LOGFILE GROUP 3;
```

如果尚未归档重做日志组，则必须指定 UNARCHIVED 关键字来清除日志文件。这用于强制您确认可能存在依赖于该重做日志进行恢复的备份，并且您决定放弃该恢复机会。这可能是您所需要的，尤其是在纠正重做日志组问题后立即执行其它备份，不再需要该重做日志文件时。

可能需要使用重做日志才能恢复当前处于脱机状态下的数据文件。

丢失了索引表空间后进行恢复

- 可以在不执行 RECOVER 任务的情况下恢复仅包含索引的表空间。
- 如果属于仅包含索引的表空间的数据文件丢失，则更为简单的方法可能是重新创建表空间和重新创建索引。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

丢失了索引表空间后进行恢复

索引是计算所得的对象，因为索引不提供任何原始数据，仅是已存在数据的另一种表示。因此，在大多数情况下，可以很容易地重新创建索引。如果您的表空间仅包含索引，则可以简化在丢失了属于该表空间的数据文件后的恢复工作。

如果丢失了此类数据文件，则可以执行以下步骤：

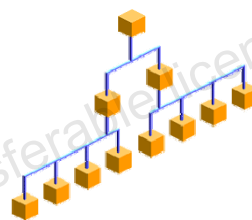
1. 删除数据文件。
2. 删除表空间。
3. 重新创建索引表空间。
4. 重新创建包含在表空间中的索引。

重新创建索引

使用以下选项可缩短重新创建索引所花费的时间：

- PARALLEL
- NOLOGGING

```
SQL> CREATE INDEX rname_idx
2 ON hr.regions (region_name)
3 PARALLEL 4;
```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

重新创建索引

创建或重新创建索引时，可以使用以下关键字来缩短创建时间：

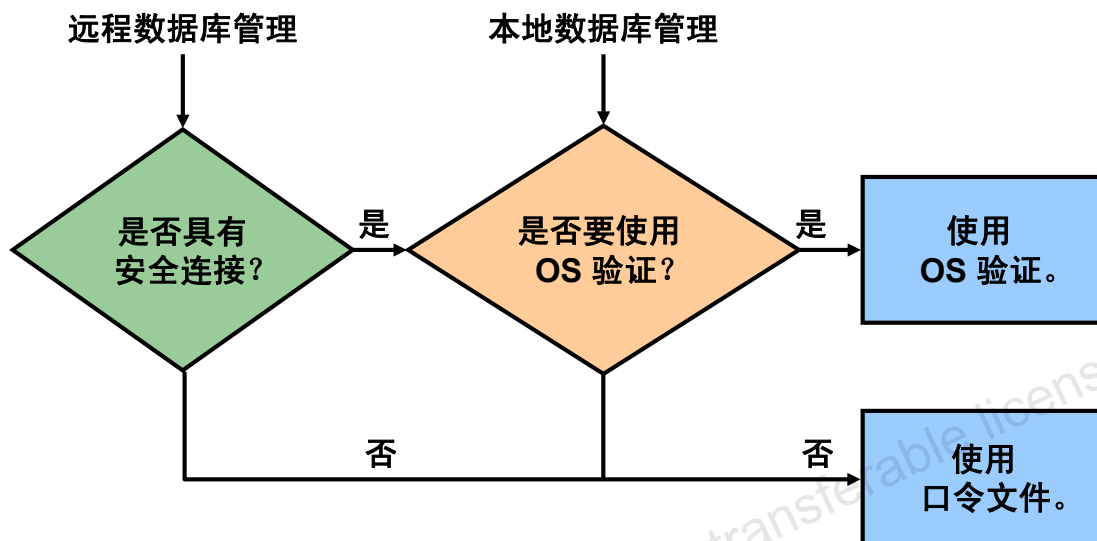
- **PARALLEL** (**NOPARALLEL** 是默认值)：多个进程可以同时协同工作来创建索引。与单个服务器进程按顺序创建索引相比，通过在多个服务器进程之间分配创建索引所需的工作，Oracle Server 可以更快速地创建索引。将随机对表取样并找到一组索引关键字，这些索引关键字按照指定的并行度将索引平均分为相同数目的片段。第一组查询进程将扫描表，提取关键字和行的 ID 对并基于关键字将每个对发送到第二组查询进程中的一个进程中。第二组中的每个进程都对关键字进行排序并按常规方式构建索引。所有索引片段构建完成后，并行协调程序会将这些片段（已进行排序）级联以形成最终的索引。
- **NOLOGGING**：使用此关键字会加快索引的创建速度，因为创建进程创建的重做日志条目极少。这种工作量大幅减小的重做生成也适用于直接路径插入和 Direct Loader (SQL*Loader) 插入。这是永久性属性，因此将显示在数据字典中。可以随时使用 ALTER INDEX NOLOGGING/LOGGING 命令来加以更新。

注：可以覆盖 NOLOGGING（如果您在数据库级别或表空间级别使用 Data Guard 或 FORCE LOGGING）。

重新创建索引（续）

索引丢失时，更为快速、简单的方法是重新创建而不是尝试恢复索引。您可以使用具有 `CONTENT=METADATA_ONLY` 参数的数据泵导出实用程序来创建包含重新创建索引的 SQL 命令的转储文件。您也可以针对之前创建的转储文件使用具有 `SQLFILE=<filename>` 参数的数据泵导入实用程序。数据泵导出和导入实用程序在《Oracle Database 11g: 数据库管理 — 课堂练习 I》课程中进行详细说明。其它信息可在《Oracle Database Utilities》中找到。

面向数据库管理员的验证方法



版权所有 © 2011, Oracle。保留所有权利。

面向数据库管理员的验证方法


根据是希望在数据库所在的那一台计算机上本地管理数据库，还是希望从一个远程客户机管理许多不同的数据库服务器，可以选择使用操作系统验证或口令文件验证来验证数据库管理员：

- 如果数据库具有口令文件且您已经具有 SYSDBA 或 SYSOPER 系统权限，则可以通过口令文件进行验证。
- 如果服务器未使用口令文件，或者您不具有 SYSDBA 或 SYSOPER 权限因而不在于口令文件中，则可以使用操作系统验证。在大多数操作系统中，数据库管理员的验证需要将数据库管理员的操作系统用户名放置到一个特殊组中，一般称为 OSDBA。该组中的用户将被授予 SYSDBA 权限。一个类似的组 OSOPER 用于向用户授予 SYSOPER 权限。

操作系统验证优先于口令文件验证。特别是，如果您是操作系统的 OSDBA 或 OSOPER 组的成员，并且以 SYSDBA 或 SYSOPER 身份连接，则连接时您将具有相关的管理权限，而与您指定的用户名/口令无关。

重新创建口令验证文件

```
SQL> grant sysdba to admin2;  
grant sysdba to admin2  
*  
ERROR at line 1:  
ORA-01994: GRANT failed: password file missing or disabled
```



在丢失了口令文件后进行恢复：

1. 使用 orapwd 重新创建口令文件。

```
$ orapwd file=$ORACLE_HOME/dbs/orapworcl password=ora entries=5
```

2. 向口令文件添加用户并向每个用户分配适当的权限。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

重新创建口令验证文件

Oracle DB 提供了一个口令实用程序 orapwd 来创建口令文件。使用 SYSDBA 权限连接时，是以 SYS 方案连接，而不是以与用户名关联的方案连接。对于 SYSOPER，则将连接到 PUBLIC 方案。使用口令文件访问数据库的权限由授权用户发出的 GRANT 命令提供。

通常，口令文件不包含在备份中，因为几乎在所有情况下，均可方便地重新创建口令文件。

保护口令文件以及标识口令文件位置的环境变量对于系统安全性是至关重要的。对这些文件和环境变量具有访问权限的任何用户都可能潜在地影响连接的安全性。

如果使用 REMOTE_LOGIN_PASSWORDFILE=EXCLUSIVE 或 SHARED 装载了数据库或实例，则不应删除或修改口令文件。否则，将无法使用该口令文件从远程重新连接。

注：口令区分大小写，在重新创建口令文件时必须考虑到这一点。而且，如果通过 IGNORECASE=Y 选项创建了原始口令文件，则必须使用相同的选项重新创建口令文件。

重新创建口令验证文件（续）

使用口令文件

下面是重新创建口令文件的步骤：

1. 使用口令实用程序 `orapwd` 创建口令文件。

```
orapwd file=filename password=password entries=max_users
```

其中：

- **filename** 是口令文件的名称（必需）。
- **password** 是 SYS 的口令（可选）。如果未包括 `password` 参数，则会提示您输入口令。
- **entries** 是允许以 SYSDBA 或 SYSOPER 身份连接的不同用户的最大数量。如果超出了此数值，必须创建新口令文件。使用较大的数值比较保险。“等号” (=) 字符两边没有空格。

示例：`orapwd file=$ORACLE_HOME/dbs/orapwU15
password=admin entries=5`

2. 使用在步骤 1 中创建的口令文件连接到数据库，并根据需要授予权限。

```
SQL> CONNECT sys/admin AS SYSDBA  
SQL> grant sysdba to admin2;
```

口令文件位置

UNIX: `$ORACLE_HOME/dbs`

Windows: `%ORACLE_HOME%\database`

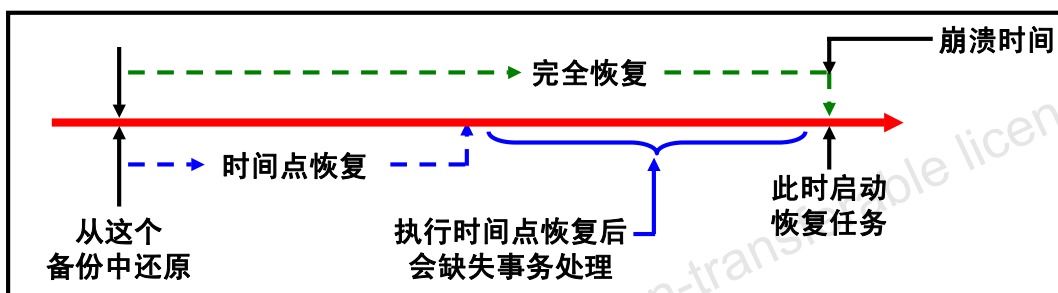
维护口令文件

使用操作系统命令删除现有口令文件，然后使用口令实用程序创建新口令文件。

比较完全恢复和不完全恢复

恢复有两类作用域：

- 完全恢复：将数据库恢复到当前最新状态，包括直至请求恢复时进行的所有已提交的数据更改
- 不完全或时间点恢复：将数据库恢复到请求恢复操作之前指定的过去时间点



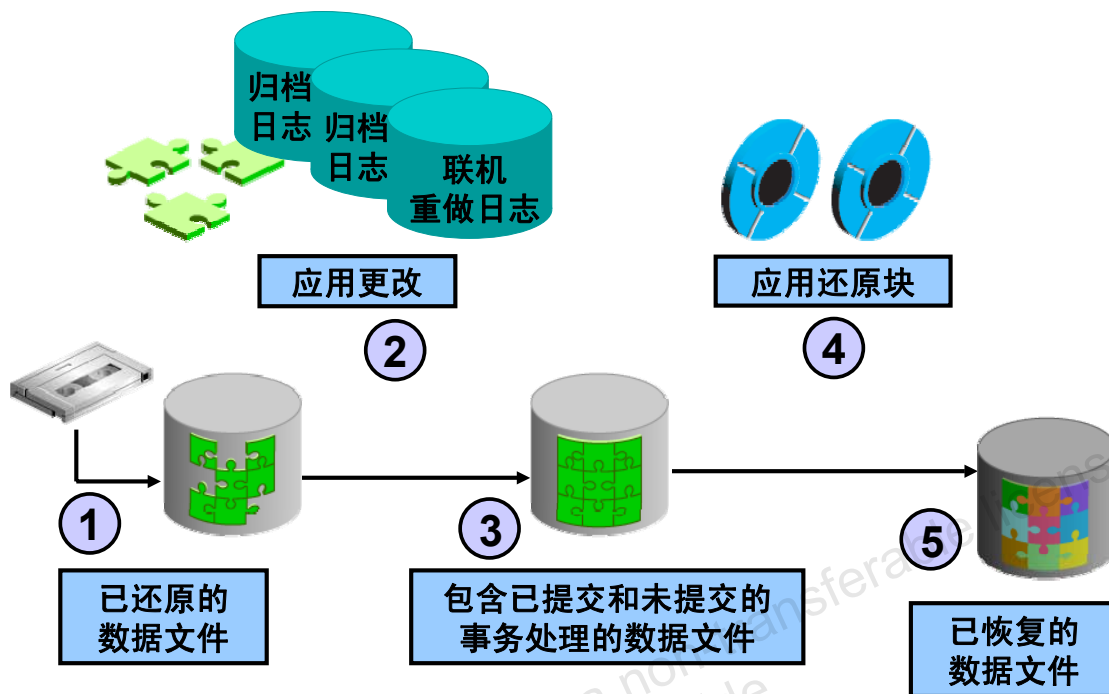
ORACLE

版权所有 © 2011, Oracle。保留所有权利。

比较完全恢复和不完全恢复

执行完全恢复时，数据库会完全恢复到最新状态，包括到当前为止提交的所有数据修改。但是执行不完全恢复时，数据库会恢复到过去时间点的某个点。它还被称为“数据库时间点恢复”。这意味着会缺失一些事务处理；即恢复目标时间和当前时间之间所做的所有数据修改都会丢失。在很多情况下，这正是想要的结果，因为可能需要撤消对数据库进行的一些更改。恢复到过去的某一时间点是删除误更改的一种方法。

完全恢复过程



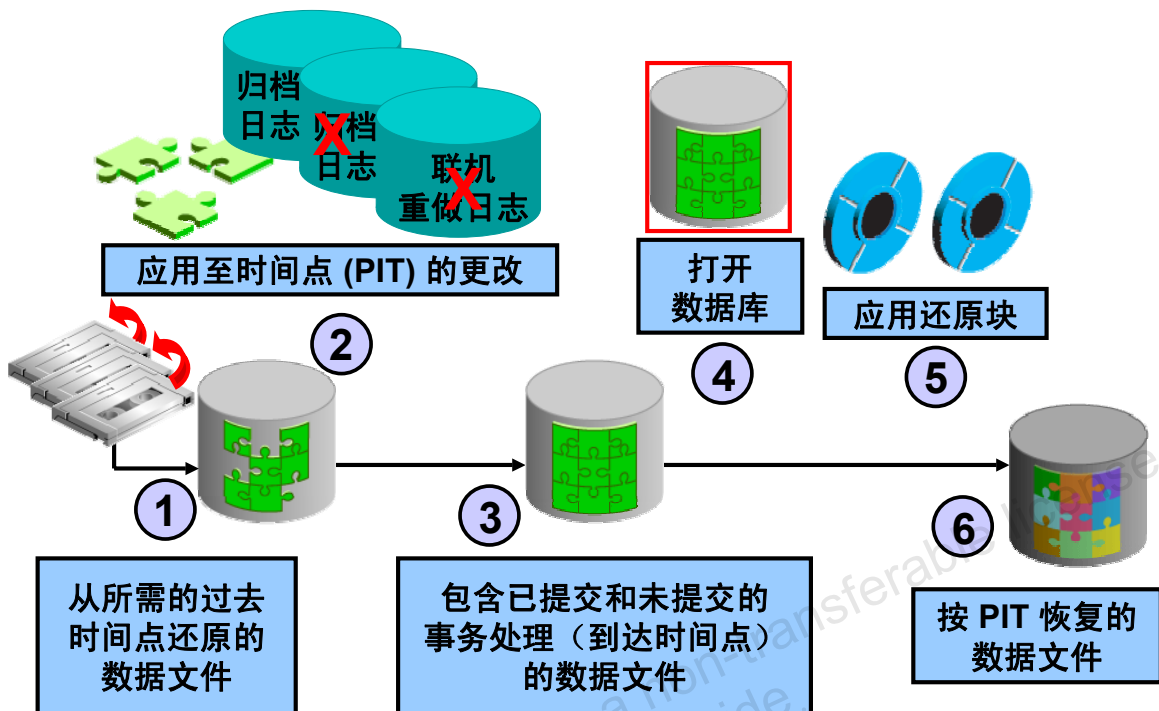
版权所有 © 2011, Oracle. 保留所有权利。

完全恢复过程

下面的步骤说明了执行完全恢复期间要采取的操作：

1. 通过备份还原损坏或缺失的文件。
2. 根据需要应用增量备份、归档重做日志文件和联机重做日志文件中的更改。将重做日志更改应用于数据文件，直到到达当前联机日志，并且重新输入了最新的事务处理。在整个过程中会生成还原块。这称为前滚或高速缓存恢复。
3. 此时，还原的数据文件中包含已提交和未提交的更改。
4. 还原块用于回退任何未提交的更改。有时也称为事务处理恢复。
5. 此时，数据文件处于已恢复状态，且与数据库中的其它数据文件一致。

时间点恢复过程



版权所有 © 2011, Oracle。保留所有权利。

时间点恢复过程

不完全恢复或数据库时间点恢复使用备份来生成非当前版本的数据库。也就是说，不会应用最新备份之后生成的所有重做记录。仅当绝对必要时才执行此类恢复。要执行时间点恢复，您需要：

- 恢复点之前生成的所有数据文件的有效脱机或联机备份
- 从备份时间到指定恢复时间的所有归档日志

下面列出了执行时间点恢复的过程：

1. **从备份还原数据文件：**如果还原点目标并不很新，则使用的备份可能也不是最新的。这需要使用 OS 命令或 RMAN RESTORE 命令来复制文件。
2. **使用 RECOVER 命令：**从归档重做日志文件应用重做，根据需要包括尽可能多的数据以达到还原点目标。
3. **恢复以后的状态：**此时，数据文件包含一些已提交的事务处理和未提交的事务处理，因为重做可以包含未提交的数据。
4. **使用 ALTER DATABASE OPEN 命令：**应用还原之前数据库已打开。这是为了提供更高的可用性。

时间点恢复过程（续）

5. **应用还原数据：**应用重做时，同时会应用支持还原数据文件的重做。这样，还原可以应用于数据文件，以便还原任何未提交的事务处理。这是接下来要完成的操作。
6. **过程完成：**此时，数据文件已恢复到所选择的时间点。

如果必须执行恢复且发现包含事务处理的归档日志丢失，其中的事务处理是在还原所用的备份的创建时间与目标恢复 SCN 之间发生的，则时间点恢复是唯一的选择。没有丢失的日志，则没有该期间内对数据文件进行更新的记录。唯一的选择就是从还原备份的时间点恢复数据库，直到未损坏的归档日志系列所允许的时间点，然后使用 RESETLOGS 选项打开数据库。缺失的重做日志文件中的或之后的所有更改都将丢失。

恢复只读表空间

对于只读表空间，在进行用户管理的备份和恢复时需要注意以下特殊事项：

- 不必为了创建数据文件的副本而将其置于备份模式。
- 创建表空间或数据文件的副本之前，不必使其脱机。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复只读表空间

由于不会向只读表空间写入数据，因此要考虑一些可使恢复过程更快更有效的特殊注意事项。将只读表空间复制到备份位置之前，不必将其置于备份模式或使其脱机。只需复制只读表空间。

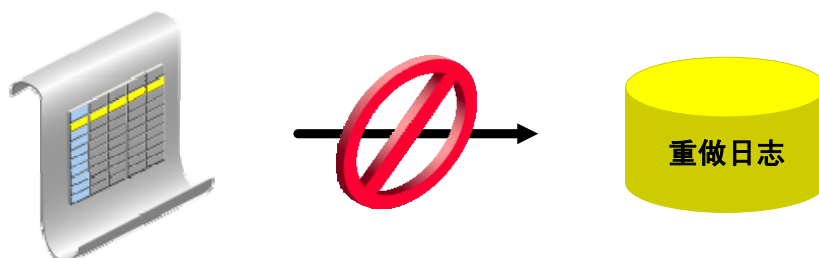
还原只读表空间时，需要使该表空间脱机、还原属于该表空间的数据文件，然后使该表空间重新联机。

请考虑下面这种将只读表空间更改为读/写状态的情况：

1. 创建只读表空间的备份。
2. 将该表空间置于读写状态。
3. 恢复该表空间。

在步骤 1 中创建的备份仍可用于恢复此表空间，即使自执行备份以来，表空间已被置于读写状态且可能已在其中写入数据。在这种情况下，从这类备份还原文件后，需要恢复表空间。

恢复 NOLOGGING 数据库对象



```
SQL> CREATE TABLE sales_copy NOLOGGING;  
SQL> INSERT /*+ APPEND */ INTO sales_copy  
2 SELECT * FROM sales_history;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复 NOLOGGING 数据库对象

利用表和索引的 NOLOGGING 属性提高效率（如果可以）。创建表作为 NOLOGGING 时，会向重做流中写入极少的重做数据，以支持对象的创建。这对于加快大型插入非常有用。在幻灯片的示例中，创建 SALES_COPY 表作为 NOLOGGING 表。因此，在按照 APPEND 提示完成插入时，不会针对特定的插入语句生成重做。因此，您无法在 SALES_HISTORY 表中恢复此事务处理。如果这是个问题，应立即对以这种方式填充的表创建备份，这一点非常重要。然后，您可以转到该表的最新备份。

如果执行介质恢复，并且涉及到 NOLOGGING 对象，则在恢复过程中会将这些对象标记为逻辑损坏。在这种情况下，请删除 NOLOGGING 对象并重新创建这些对象。

使用 REPORT UNRECOVERABLE RMAN 命令列出任何包含一个或多个对象的表空间名称，从该表空间的最新备份起，已为表空间中的对象执行 NOLOGGING 操作。

在丢失了所有控制文件副本后进行恢复：概览

	当前	备份
可用	还原备份控制文件，执行完全恢复，OPEN RESETLOGS	还原备份控制文件，执行完全恢复，OPEN RESETLOGS
不可用	重新创建控制文件，OPEN RESETLOGS	还原备份控制文件，执行时间点恢复，OPEN RESETLOGS

联机日志状态 ↑

数据文件状态 →

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

在丢失了所有控制文件副本后进行恢复：概览

丢失了所有的控制文件不应再发生。**预防优于恢复**。即使您将控制文件副本存储在不同位置，仍有可能需要在丢失了所有这些副本后进行恢复。如果丢失了当前控制文件的所有副本，但拥有备份控制文件，则操作过程取决于联机日志文件和数据文件的状态。幻灯片中的图表显示了每种情况下应执行的操作。

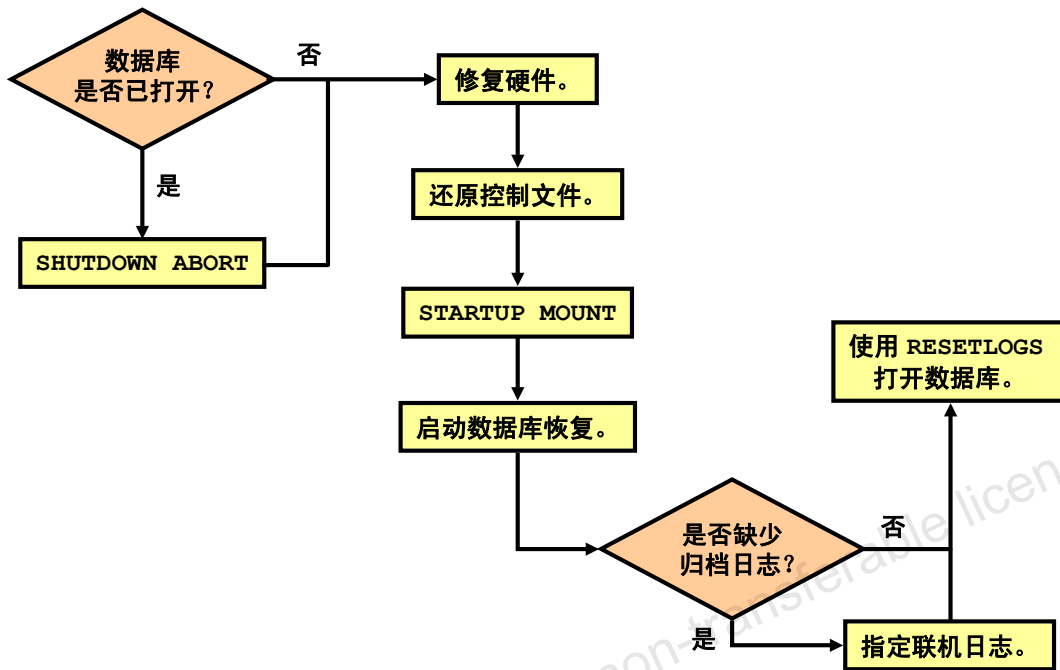
联机日志可用

如果联机日志可用且包含恢复所必需的重做，且数据文件是最新的，则可以还原备份控制文件，执行完全恢复，并使用 RESETLOGS 选项打开数据库。恢复期间，必须指定联机重做日志的文件名。如果数据文件不是最新的，请执行相同的过程。

联机日志不可用

如果联机日志不可用，但数据文件是最新的，请重新创建控制文件并打开 RESETLOGS。但是，如果数据文件不是最新的，请还原备份控制文件，执行时间点恢复，并打开 RESETLOGS。

将控制文件恢复到默认位置



版权所有 © 2011, Oracle. 保留所有权利。

将控制文件恢复到默认位置

如果需要恢复控制文件，且默认位置仍为有效位置，请按照幻灯片中所示的步骤执行操作。必须先关闭数据库，然后修复任何硬件故障，使默认位置是有效位置。将控制文件还原到默认位置。使用如下命令执行此操作，将备份控制文件复制到默认位置：

```
% cp /backup/control01.dbf /disk1/oradata/trgt/control01.dbf
% cp /backup/control02.dbf /disk2/oradata/trgt/control02.dbf
```

装载数据库，并启动恢复过程。必须指定所使用的备份控制文件。

```
SQL> RECOVER DATABASE USING BACKUP CONTROLFILE UNTIL CANCEL;
```

如果在恢复过程中，系统提示存在缺失的重做日志，这可能表示缺失的重做日志是联机重做日志文件。看到提示时，请提供联机重做日志文件的名称。恢复完成后，请打开数据库，指定 RESETLOGS 选项。

（有关该主题的更多内容，请查看下一课。）

测验

以下哪种情况可能会发出 RMAN RECOVER 命令？

1. 数据库在 NOARCHIVELOG 模式下使用完全备份。
2. 数据库在 ARCHIVELOG 模式下使用完全备份。
3. 数据库在 NOARCHIVELOG 模式下使用增量备份。
4. 数据库在 ARCHIVELOG 模式下使用增量备份。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2、3、4

测验

您丢失了口令文件，则作为 DBA，您可从何处恢复条目，从而使您可重新创建丢失的口令文件？

1. 仅从 RMAN 目录
2. 从控制文件
3. 从 Oracle Enterprise Manager 资料档案库
4. 从数据字典
5. 您必须重新手动授予 SYSOPER、SYSDBA 和 SYSASM 条目权限

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：5

小结

在本课中，您应该已经学会：

- 描述文件丢失的原因并确定相应的操作
- 描述主要的恢复操作
- 备份和恢复控制文件
- 在丢失了重做日志组后进行恢复

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 RMAN 执行恢复

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

学完本课后，应能使用 RMAN 完成以下工作：

- 在丢失关键或非关键数据文件后执行完全恢复
- 使用增量更新的备份进行恢复
- 切换到映像副本进行快速恢复
- 将数据库还原到新主机上
- 使用备份控制文件进行恢复

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

课程目标

有关更多详细资料，请参阅《Oracle Database Backup and Recovery User's Guide》。

使用 RMAN RESTORE 和 RECOVER 命令

- RESTORE 命令：从备份中还原数据库文件
- RECOVER 命令：通过应用增量备份和重做日志文件中记录的更改来恢复已还原文件

```
RMAN> SQL 'ALTER TABLESPACE inv_tbs OFFLINE IMMEDIATE';
RMAN> RESTORE TABLESPACE inv_tbs;
RMAN> RECOVER TABLESPACE inv_tbs;
RMAN> SQL 'ALTER TABLESPACE inv_tbs ONLINE';
```

- Oracle Enterprise Manager 恢复向导创建并运行一个 RMAN 脚本以执行恢复

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 RMAN RESTORE 和 RECOVER 命令

从备份中重建整个数据库或数据库某一部分的过程通常包含两个阶段：从备份中检索数据文件的副本，以及从归档和联机重做日志中重新应用自备份以来对文件所做的更改，以使数据库恢复到所需的 SCN（通常为最新的 SCN）。

- RESTORE {DATABASE | TABLESPACE name [,name]... | DATAFILE name [,name] }...

RESTORE 命令将数据文件从磁带、磁盘或其它介质上的备份位置检索到磁盘上，并使其可供数据库服务器使用。RMAN 会从备份中还原恢复操作期间所需的任何归档重做日志。如果备份存储在介质管理器上，则必须配置或分配用于访问存储在介质上的备份的通道。

- RECOVER {DATABASE | TABLESPACE name [,name]... | DATAFILE name [,name] }...

RECOVER 命令获取已还原的数据文件副本，并将增量备份和数据库重做日志中记录的更改应用于该副本。

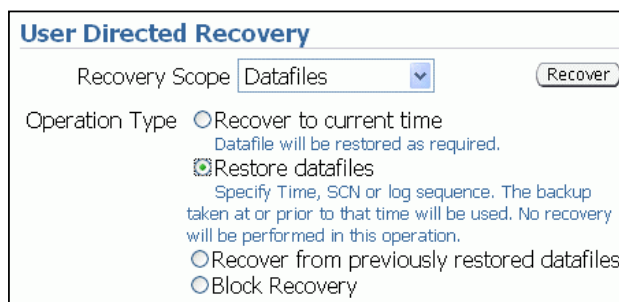
还可以通过 Oracle Enterprise Manager 使用恢复向导来执行完全恢复或时间点恢复。在“Availability（可用性）”页上，在“Backup/Recovery（备份/恢复）”部分中单击“Perform Recovery（执行恢复）”。

使用 RMAN RESTORE 和 RECOVER 命令（续）

注：一种自动检测恢复需求并执行该恢复的方法是使用数据恢复指导，这将在“诊断数据库”一课中进行介绍。

执行完全恢复：在 ARCHIVELOG 模式下 丢失了非关键数据文件

如果某个数据文件丢失或损坏，且该文件不属于 SYSTEM 或 UNDO 表空间，则只还原并恢复缺失的数据文件。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行完全恢复：在 ARCHIVELOG 模式下丢失了非关键数据文件

当数据库处于 ARCHIVELOG 模式时，如果丢失了任何不属于 SYSTEM 或 UNDO 表空间的数据文件，则只会影响缺失文件中的对象。

要使用 Oracle Enterprise Manager 还原并恢复缺失的数据文件，请按照以下步骤执行操作：

1. 在“Availability（可用性）”属性页上，单击“Perform Recovery（执行恢复）”。
2. 选择“Datafiles（数据文件）”作为“Recovery Scope（恢复范围）”，并选择“Restore datafiles（还原数据文件）”作为“Operation Type（操作类型）”。
3. 添加需要恢复的所有数据文件。
4. 指定从哪个备份还原文件。
5. 确定是将文件还原至默认位置还是新位置（如果磁盘或控制器缺失）。
6. 提交 RMAN 作业来还原和恢复缺失的文件。

由于数据库处于 ARCHIVELOG 模式，所以可恢复到最后提交的时间，用户不需要重新输入任何数据。

执行完全恢复：在 ARCHIVELOG 模式下 丢失了系统关键数据文件

如果某个数据文件丢失或损坏，且该文件属于 SYSTEM、UNDO（或 SYSAUX）表空间，请执行以下步骤：

1. 实例可能会也可能不会自动关闭。如果未自动关闭，请使用 SHUTDOWN ABORT 关闭实例。
2. 装载数据库。
3. 还原并恢复缺失的数据文件。
4. 打开数据库。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行完全恢复：在 ARCHIVELOG 模式下丢失了系统关键数据文件

属于 SYSTEM 表空间或包含 UNDO 数据的数据文件被视为系统关键数据文件。如果使用 Oracle Enterprise Manager 进行恢复，则 SYSAUX 表空间也很关键。如果丢失了这些文件中的一个，就需要从 MOUNT 状态还原数据库（不同于可以在数据库处于打开状态时还原的其它数据文件）。

请通过执行以下步骤进行完全恢复：

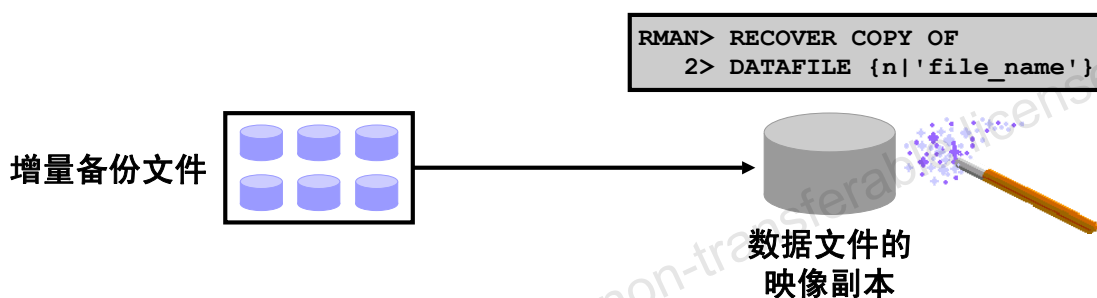
1. 如果实例尚未关闭，请关闭实例。
2. 装载数据库。
3. 在“Maintenance（维护）”属性页上，单击“Perform Recovery（执行恢复）”。
4. 选择“Datafiles（数据文件）”作为恢复类型，然后选择“Restore to current time（还原到当前时间）”。
5. 添加需要恢复的所有数据文件。
6. 确定要将文件还原到默认位置还是新位置（如果磁盘或控制器缺失）。
7. 提交 RMAN 作业来还原和恢复缺失的文件。
8. 打开数据库。用户不需要重新输入数据，因为将恢复到最后一次提交的时间。

注：数据恢复指导会检测这种恢复情形，这将在“诊断数据库”一课中进行介绍。

恢复映像副本

RMAN 可使用增量备份来恢复映像副本：

- 映像副本会使用截至增量备份 SCN 的所有更改来进行更新。
- 增量备份减少了介质恢复所需的时间。
- 增量还原之后不需要执行映像副本。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复映像副本

可使用 RMAN 将增量备份应用于数据文件映像副本。在这种恢复方法中，您使用 RMAN 恢复数据文件的副本，即通过将增量备份应用于映像副本将映像副本前滚（恢复）至指定的时间点。映像副本使用执行增量备份时的 SCN 之前的所有更改来进行更新。RMAN 使用在介质恢复中得到的更新数据文件，如同使用在该 SCN 得到的完全映像副本一样，因此没有每天创建数据库完全映像副本所需的开销。下面介绍了将增量备份应用于数据文件映像副本的优点：

- 由于您只需要应用自上一次增量备份以来的归档日志，因此可以减少介质恢复（使用归档日志）所需的时间。
- 在增量还原之后不需要执行完全映像副本。

如果在应用增量备份文件期间恢复过程失败，只需要重新启动恢复过程。在创建映像数据文件副本之前到要停止恢复过程之前的这段时间内，RMAN 自动确定这期间所需应用的增量备份文件。如果 RMAN 目录中记录了多个版本的映像副本，RMAN 自动使用最新版本的映像副本。如果 RMAN 不能将增量备份文件与映像副本合并，则它会报告错误。

恢复映像副本：示例

如果每天都运行以下命令：

```

RMAN> recover copy of database with tag 'daily_inc';
RMAN> backup incremental level 1 for recover of copy
2> with tag 'daily_inc' database;
  
```

结果如下：

	RECOVER	BACKUP
第 1 天	无	创建映像副本
第 2 天	无	创建 1 级增量备份
第 3 天和以后	根据增量备份恢复副本	创建 1 级增量备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复映像副本：示例

如果每天都运行幻灯片中所示的命令，则可以随时获得所有数据库数据文件的连续更新的映像副本。

图表显示了每次运行所发生的操作。请注意，这种算法需要一段准备期间；到第 3 天后策略才会起作用。

第 1 天： RECOVER 命令不执行任何操作。尚不存在要恢复的映像副本。BACKUP 命令创建映像副本。

第 2 天： RECOVER 命令仍然不执行任何操作。因为尚不存在增量备份。由于已在第 1 天创建了基线映像副本，BACKUP 命令会创建增量备份。

第 3 天： RECOVER 命令将增量备份中的更改应用于映像副本。BACKUP 命令将执行另一个增量备份，该备份将在第 4 天用于恢复映像副本。依此进行循环。

实施这种备份策略时一定要使用标记，这一点非常重要。标记可以将这些特定的增量备份链接至所创建的映像副本。如果不使用标记，则可能会使用最新但可能不正确的增量备份来恢复映像副本。

执行到映像副本的快速切换

通过执行以下步骤来执行快速恢复：

1. 使数据文件脱机。
2. 使用 SWITCH TO ... COPY 命令切换到映像副本。
3. 恢复数据文件。
4. 使数据文件联机。

现在，数据文件已恢复并且可以在新的位置中使用。

也可选择执行以下操作，将这些文件放回其原始位置：

5. 在原始位置创建数据文件的映像副本。
6. 使数据文件脱机。
7. SWITCH TO ... COPY。
8. 恢复数据文件。
9. 使数据文件联机。

```
SQL> SWITCH DATAFILE 'filename' TO COPY;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行到映像副本的快速切换

执行以下步骤，可以使用数据文件的映像副本进行快速恢复：

1. 使数据文件脱机。
2. 使用 SWITCH TO ... COPY 命令指向这些文件的映像副本。
3. 恢复数据文件。
4. 使数据文件联机。

此时，数据库是可用的，且数据文件已恢复。但是，如果您希望将数据文件放回其原始位置，请继续执行以下步骤：

5. 使用 BACKUP AS COPY 命令在原始位置创建数据文件的映像副本。
6. 使数据文件脱机。
7. 使用 SWITCH TO COPY 命令切换到在步骤 5 中创建的副本。
8. 恢复数据文件。
9. 使数据文件联机。

可以使用此命令来恢复数据文件、表空间、临时文件或整个数据库。切换到的目标文件必须为映像副本。

使用 SET NEWNAME 切换文件

- 在 RUN 块中使用 SET NEWNAME 命令来还原到非默认位置。

```

RUN
{ ALLOCATE CHANNEL dev1 DEVICE TYPE DISK;
  ALLOCATE CHANNEL dev2 DEVICE TYPE sbt;
  SQL "ALTER TABLESPACE users OFFLINE IMMEDIATE";
  SET NEWNAME FOR DATAFILE '/disk1/oradata/prod/users01.dbf'
    TO '/disk2/users01.dbf';
  RESTORE TABLESPACE users;
  SWITCH DATAFILE ALL;
  RECOVER TABLESPACE users;
  SQL "ALTER TABLESPACE users ONLINE";
}

```

- 请为数据库或已命名表空间中的所有文件指定一个默认的名称格式，不要单独指定各个名称。
- 默认名称用于 RUN 块中的 DUPLICATE、RESTORE 和 SWITCH 命令。

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

使用 SET NEWNAME 切换文件

SET NEWNAME 命令只能在 RUN 块中使用。该命令可以为后续操作准备名称映射。在幻灯片中的示例中，SET NEWNAME 命令定义了该数据文件的还原操作的写入位置。执行 RESTORE 命令后，users01.dbf 数据文件将还原到 /disk2/users01.dbf。数据文件将写入到该位置，但控制文件仍不指向该位置。SWITCH 命令将导致使用新位置更新控制文件。

更有效的方法是，使用 SET NEWNAME 子句为已命名表空间中的所有数据文件和数据库中的所有数据文件指定默认的名称格式（而不是像在 Oracle Database 11gR2 (11.2) 之前的数据库版本中那样分别设置文件名）。

SET NEWNAME 命令的优先顺序如下所示：

1. SET NEWNAME FOR DATAFILE 和 SET NEWNAME FOR TEMPFILE
2. SET NEWNAME FOR TABLESPACE
3. SET NEWNAME FOR DATABASE

SET NEWNAME 的替代变量

语法元素	说明
%b	指定不带目录路径的文件名 *新增*
%f	指定为其生成新名称的数据文件的绝对文件号
%I	指定 DBID
%N	指定表空间名称
%U	指定如下格式的系统生成文件名： data-D-%d_id-%I_TS-%N_FNO-%f
RUN { SET NEWNAME FOR DATAFILE 1 TO '/oradata1/system01.dbf'; SET NEWNAME FOR DATAFILE 2 TO '/oradata2/sysaux01.dbf'; SET NEWNAME FOR DATAFILE 3 TO '/oradata3/undotbs01.dbf'; SET NEWNAME FOR DATAFILE 4 TO '/oradata4/users01.dbf'; SET NEWNAME FOR TABLESPACE example TO '/oradata5/%b'; DUPLICATE TARGET DATABASE TO dupldb; }	

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

SET NEWNAME 的替代变量

要在还原到其它位置时避免可能的名称冲突，请使用 SET NEWNAME 命令的替代变量。请至少指定以下替代变量中的一种：%b、%f 和 %U。%I 和 %N 是可选变量。

该示例显示 SET NEWNAME FOR TABLESPACE 命令使用替代变量和显式 SET NEWNAME 子句来设置默认名称。

在 NOARCHIVELOG 模式下 执行数据库还原和恢复

- 当数据库处于 NOARCHIVELOG 模式时，如果丢失了任何数据文件，请执行以下任务：
 - 如果实例尚未关闭，请关闭实例。
 - 从备份还原整个数据库，包括所有数据文件和控制文件。
 - 打开数据库。
- 用户必须重新输入自上一次备份以来所做的更改。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

在 NOARCHIVELOG 模式下执行数据库还原和恢复

如果在 NOARCHIVELOG 模式下丢失了数据库中的任何数据文件，则需要完全还原数据库，包括控制文件和所有数据文件。如果具有增量备份，则需要执行还原和恢复操作。如果丢失的数据文件属于只读表空间，则只需还原该文件。

当数据库处于 NOARCHIVELOG 模式时，只能恢复到上一次备份时的状态。因此，用户必须重新输入自上一次备份以来所做的更改。

对于该类型的恢复，请使用 RESTORE 和 RECOVER 命令，或者在 Oracle Enterprise Manager 中执行以下任务：

1. 如果实例尚未关闭，请关闭实例。
2. 在“Maintenance（维护）”属性页上，单击“Perform Recovery（执行恢复）”。
3. 选择“Whole Database（整个数据库）”作为恢复类型。

使用还原点

还原点为时间点提供了名称：

- 现在：

```
SQL> CREATE RESTORE POINT before_mods;
```

- 过去某个时间：

```
SQL> CREATE RESTORE POINT end_q1 AS OF SCN 100;
```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用还原点

可以为特定时间点或 SCN 编号指定名称。这可以在将来执行时间点恢复或闪回操作时用作参考。

- 幻灯片中的第一个示例创建了一个表示当前时间点的还原点。如果您打算在数据库中应用对应用程序或数据的更新，并且希望反向引用数据库的此状态，则可以使用 BEFORE_MODS 还原点。
- 幻灯片中的第二个示例创建了一个表示过去的 SCN 100 的还原点。此还原点的使用方式与上一个还原点相同。

正常情况下，还原点至少在数据库中保留由 CONTROL_FILE_RECORD_KEEP_TIME 初始化参数指定的时间长度。不过，在创建还原点时可以使用 PRESERVE 选项，该选项会使还原点一直保存到被显式删除为止。

可以在 V\$RESTORE_POINT 视图中查看还原点的名称、SCN、时间戳和其它信息。

执行时间点恢复

通过执行以下操作来执行服务器管理的时间点恢复：

1. 确定还原目标点：SCN、时间、还原点或日志序列号。
2. 相应地设置 NLS 环境变量。
3. 装载数据库。
4. 使用 SET UNTIL、RESTORE 和 RECOVER 命令准备并运行 RUN 块。
5. 在 READONLY 模式下打开数据库，并验证恢复点是否为您需要的恢复点。
6. 使用 RESETLOGS 打开数据库。



ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

执行时间点恢复

可以通过以下步骤执行服务器管理的时间点恢复。数据库必须处于 ARCHIVELOG 模式。

1. 确定还原目标。这可以是日期和时间、SCN、还原点或日志序列号。例如，如果您知道某些错误事务处理是在昨天下午 3:00 提交的，则可以选择昨天下午 2:59 作为目标还原点时间。
2. 设置国家语言支持 (NLS) 操作系统环境变量，确保为 RMAN 提供的时间常量的格式是正确的。以下是一些示例设置：


```
$ export NLS_LANG = american_america.us7ascii
$ export NLS_DATE_FORMAT = "yyyy-mm-dd:hh24:mi:ss"
```
3. 装载数据库。如果数据库已打开，则必须先将其关闭，如本例所示：


```
RMAN> shutdown immediate
RMAN> startup mount
```

执行时间点恢复（续）

4. 创建一个 RUN 块并运行该块。RECOVER 和 RESTORE 命令应位于同一个 RUN 块中，这样 UNTIL 设置可以同时应用于两者。例如，如果选择恢复到特定 SCN，则 RESTORE 命令需要知道该值，以便可以从足够早的备份（即该 SCN 之前的备份）还原文件。以下是 RUN 块的一个示例：

```
RUN
{
    SET UNTIL TIME '2007-08-14:21:59:00';
    RESTORE DATABASE;
    RECOVER DATABASE;
}
```

5. 打开数据库进行读/写操作时，会立即完成刚刚执行的还原。因此，请先在 READ ONLY 模式下打开数据库并查看某些数据，检查恢复操作是否符合预期结果。

```
RMAN> SQL 'ALTER DATABASE OPEN READ ONLY';
```

6. 如果对恢复结果感到满意，请使用 RESETLOGS 选项打开数据库，如下所示：

```
RMAN> ALTER DATABASE OPEN RESETLOGS;
```

使用备份控制文件执行恢复

- 当前控制文件的所有副本都丢失或损坏时还原和装载备份控制文件。
- 还原备份控制文件之后执行 RECOVER 命令。
- 执行完全恢复或时间点恢复后使用 RESETLOGS 选项打开数据库。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用备份控制文件执行恢复

如果丢失了当前控制文件的所有副本，则在执行恢复之前必须还原和装载备份控制文件。通过执行恢复操作可恢复丢失的数据文件，也可只恢复控制文件。如果使用了恢复目录，这个过程与使用当前控制文件进行恢复的过程是完全一样的，因为 RMAN 可使用恢复目录获取 RMAN 元数据。

恢复丢失的服务器参数文件

使用 FROM MEMORY 子句可以创建系统范围内的当前参数设置。

```
SQL> CREATE PFILE [= 'pfile_name' ]  
      FROM { { SPFILE [= 'spfile_name'] } | MEMORY } ;
```

```
SQL> CREATE SPFILE [= 'spfile_name' ]  
      FROM { { PFILE [= 'pfile_name' ] } | MEMORY } ;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复丢失的服务器参数文件

恢复服务器参数文件的最简单方法是使用 FROM MEMORY 子句，该子句可以使用系统范围内的当前参数设置来创建文本初始化参数文件 (PFILE) 或服务器参数文件 (SPFILE)。在 RAC 环境中，所创建的文件包含来自每个实例的参数设置。

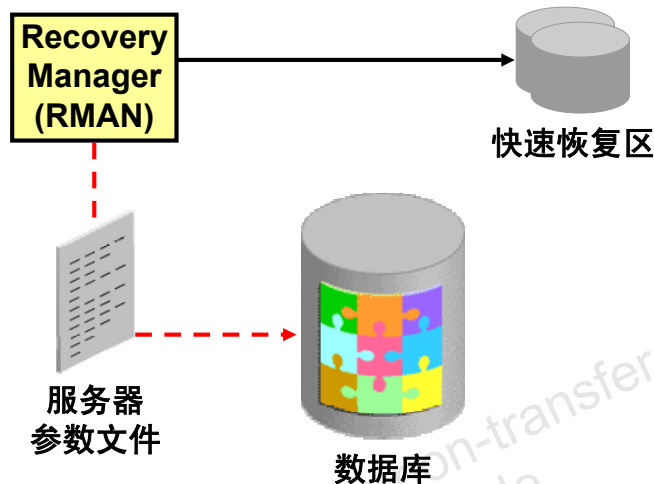
在实例启动期间，所有参数设置均记录到 alert.log 文件中。在 Oracle Database 11g 中，alert.log 参数转储文本是使用有效的参数语法编写的。这便于对参数进行剪切，并将其粘贴到单独的文件中，从而用作后续实例的 PFILE。在实例启动时，PFILE 或 SPFILE 的名称被写入到 alert.log。使用未知的客户机 PFILE 时，预警日志也会对此加以提示。要支持其它功能，必须将 COMPATIBLE 初始化参数设置为 11.0.0.0 或更高。

从控制文件自动备份还原服务器参数文件

```

RMAN> STARTUP FORCE NOMOUNT;
RMAN> RESTORE SPFILE FROM AUTOBACKUP;
RMAN> STARTUP FORCE;

```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

从控制文件自动备份还原服务器参数文件

如果服务器参数文件已丢失，并且无法使用 FROM MEMORY 子句，可以从自动备份来还原该文件。这个过程类似于从自动备份中还原控制文件。如果自动备份没有位于快速恢复区中，请为您的数据库设置 DBID。发出 RESTORE SPFILE FROM AUTOBACKUP 命令。

如果要将 SPFILE 还原到非默认位置，请按以下方式指定命令：

```
RESTORE SPFILE TO <file_name> FROM AUTOBACKUP
```

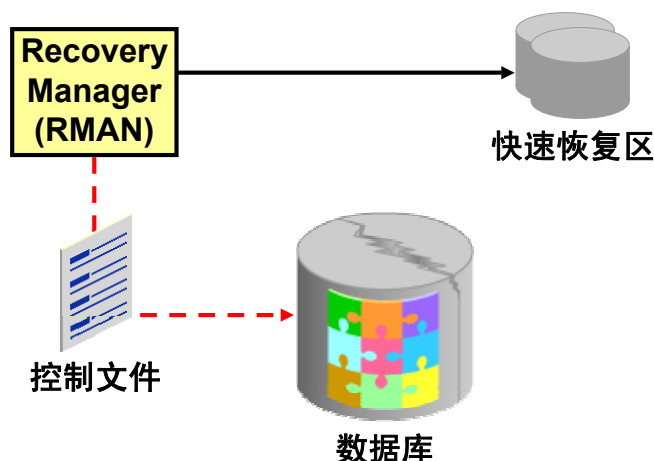
如果要从快速恢复区还原服务器参数文件，请按以下方式指定命令：

```

RMAN> run {
2> restore spfile from autobackup
3> recovery area = '<flash recovery area destination>'
4> db_name = '<db_name>';
5> }

```

从自动备份还原控制文件



```
RMAN> STARTUP NOMOUNT;
RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP;
RMAN> ALTER DATABASE MOUNT;
RMAN> RECOVER DATABASE;
RMAN> ALTER DATABASE OPEN RESETLOGS;
```

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

从自动备份还原控制文件

如果没有使用恢复目录，则应配置控制文件的自动备份，以便能够在需要时快速还原控制文件。不论是否使用快速恢复区，用于还原控制文件的命令都是相同的。但是，如果使用快速恢复区，RMAN 会隐式交叉检查控制文件中列出的备份和映像副本，并将位于快速恢复区中且已还原的控制文件中未记录的所有文件编入目录，这样可提高已还原的控制文件在还原数据库其余部分时所起的作用。

使用幻灯片中所示的命令在丢失控制文件后进行恢复。首先，在 NOMOUNT 模式下启动实例。因为没有控制文件，所以不能装载实例。从备份还原控制文件。因为存在控制文件，所以可以装载数据库。现在，必须恢复数据库，因为当前拥有包含有关旧版本数据库的信息的备份控制文件。恢复数据库后，可以将其打开。由于新的控制文件代表数据库的不同实例，因此必须指定 RESETLOGS。

注：还原控制文件后并不自动对磁带备份进行交叉检查。如果使用磁带备份，则在还原控制文件并装载数据库之后，必须对磁带上的备份进行交叉检查。

从自动备份还原控制文件（续）

要从自动备份还原控制文件，数据库必须处于 NOMOUNT 状态。如果自动备份没有位于快速恢复区中，则在发出 RESTORE CONTROLFILE FROM AUTOBACKUP 命令之前必须设置数据库标识符 (DBID)，如下例所示：

```
RMAN> SHUTDOWN ABORT;  
RMAN> STARTUP NOMOUNT;  
RMAN> SET DBID 1090770270;  
RMAN> RESTORE CONTROLFILE FROM AUTOBACKUP;
```

RMAN 搜索控制文件自动备份。如果找到了自动备份，RMAN 会将该备份的控制文件还原到 CONTROL_FILES 初始化参数中列出的所有控制文件位置。

如果有恢复目录，则不必设置 DBID，也不必使用控制文件自动备份来还原控制文件。可以使用不带参数的 RESTORE CONTROLFILE 命令：

```
RMAN> RESTORE CONTROLFILE;
```

在执行此操作时，实例必须处于 NOMOUNT 状态，且 RMAN 必须连接到恢复目录。还原的控制文件会写入 CONTROL_FILES 初始化参数中列出的所有位置。

使用 RESTORE CONTROLFILE ... TO <destination> 命令将控制文件还原到非默认位置。

如果数据库的 SPFILE 也丢失了，而且需要从自动备份中还原，则还原过程与从自动备份还原控制文件相似。必须先为数据库设置 DBID，然后再使用 RESTORE SPFILE FROM AUTOBACKUP 命令。

在使用还原的服务器参数文件启动实例后，RMAN 可从自动备份中还原控制文件。还原和装载控制文件后，就拥有了还原和恢复数据库所必需的备份信息。

从备份还原数据库的控制文件后，必须执行完全介质恢复，然后使用 RESETLOGS 选项打开数据库。

使用增量备份恢复处于 NOARCHIVELOG 模式的数据库

使用增量备份可对处于 NOARCHIVELOG 模式的数据库执行有限恢复。

```
STARTUP FORCE NOMOUNT;  
RESTORE CONTROLFILE;  
ALTER DATABASE MOUNT;  
RESTORE DATABASE;  
RECOVER DATABASE NOREDO;  
ALTER DATABASE OPEN RESETLOGS;
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用增量备份恢复处于 NOARCHIVELOG 模式的数据库

使用增量备份可对处于 NOARCHIVELOG 模式的数据库执行有限恢复。增量备份必须是一致备份。

如果创建了增量备份，RMAN 将使用 0 级和 1 级备份来还原和恢复数据库。

如果联机重做日志文件已丢失或无法应用于增量备份，则必须在 RECOVER DATABASE 命令中指定 NOREDO 选项。如果未指定 NOREDO 选项，RMAN 会在应用增量备份后搜索联机重做日志文件。如果联机重做日志文件不可用，RMAN 就会发布错误消息。

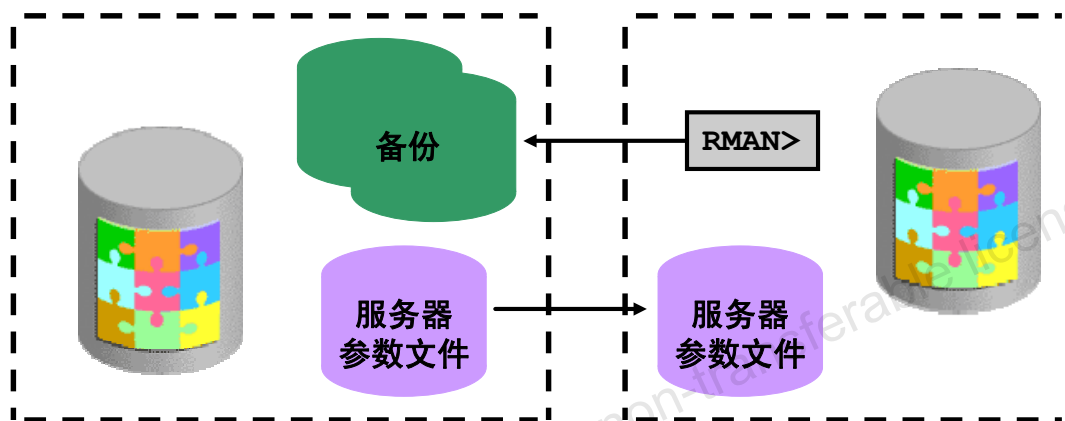
如果当前联机重做日志文件包含自上一次增量备份以来的所有更改，则可在不指定 NOREDO 选项的情况下发出 RECOVER DATABASE 命令并应用更改。

注：仅当控制文件不是当前控制文件时才需要对其进行还原。

在新主机上还原和恢复数据库

使用此过程可执行以下操作：

- 执行测试还原
- 将生产数据库移到新主机上



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

在新主机上还原和恢复数据库

使用后面的页中描述的过程执行测试还原。还可使用此过程将生产数据库移到新主机上。

已还原测试数据库的数据库标识符 (DBID) 与原始数据库的 DBID 相同。如果使用恢复目录连接测试数据库和恢复目录数据库，就会使用测试数据库的信息更新恢复目录。这会影响 RMAN 还原和恢复源数据库的能力。

如果您的目的是创建要在新主机上使用的目标数据库的新副本，则应使用 RMAN DUPLICATE 命令创建副本数据库。副本数据库分配了一个新 DBID，所以可在与原始目标数据库相同的恢复目录中注册该数据库。有关 DUPLICATE 命令的详细信息，请参阅“使用 RMAN 复制数据库”一课。

准备将数据库还原到新主机

要为还原数据库做准备，请执行以下步骤：

- 记录源数据库的数据库标识符 (DBID)。
- 将源数据库初始化参数文件复制到新主机。
- 确保在还原主机上可访问源备份，包括控制文件自动备份。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

准备将数据库还原到新主机

执行幻灯片中列出的步骤为将数据库还原到新主机做准备。

注：如果要执行测试还原，请勿在还原数据文件时连接到恢复目录。如果连接到恢复目录，RMAN 会在恢复目录中记录关于已还原数据文件的信息，还会认为已还原数据库是当前目标数据库。如果控制文件不够大，不能包含备份中需要还原的所有 RMAN 资料档案库数据，而此时又必须使用恢复目录，请导出恢复目录，然后将其导入不同的方案或数据库中。使用复制的恢复目录进行测试还原。

将数据库还原到新主机

为了还原数据库，请在还原主机上执行以下步骤：

1. 配置 ORACLE_SID 环境变量。
2. 启动 RMAN 并在 NOCATALOG 模式下连接到目标实例。
3. 设置数据库标识符 (DBID)。
4. 在 NOMOUNT 模式下启动实例。
5. 从备份集中还原服务器参数文件。
6. 关闭实例。
7. 编辑还原的初始化参数文件。
8. 在 NOMOUNT 模式下启动实例。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

将数据库还原到新主机

为了还原数据库，请在还原主机上执行本页和下页中列出的步骤。

1. 配置 ORACLE_SID 环境变量，如下例所示：

```
$ setenv ORACLE_SID orcl
```
2. 启动 RMAN 并连接到目标实例。请勿连接到恢复目录，如下所示：

```
$ rman TARGET /
```
3. 设置数据库标识符 (DBID)。通过查询 V\$DATABASE 中的 DBID 列可找到源数据库的 DBID。

```
RMAN> SET DBID 1090770270;
```
4. 在 NOMOUNT 模式下启动实例：

```
RMAN> STARTUP NOMOUNT
```

由于服务器参数文件尚未还原，所以会收到类似以下内容的错误。RMAN 会使用“虚”参数文件启动实例。

```
startup failed: ORA-01078: failure in processing system parameters
```
5. 从备份集中还原服务器参数文件并关闭实例，如下例所示：

```
RESTORE SPFILE TO PFILE '?/oradata/test/initiorcl.ora' FROM AUTOBACKUP;
```

将数据库还原到新主机（续）

6. 关闭实例：

```
SHUTDOWN IMMEDIATE;
```

7. 通过编辑还原的初始化参数文件来更改特定于位置的任何参数（如以 `_DEST` 结尾的参数）以反映新目录结构。

8. 使用编辑后的文本初始化参数文件在 NOMOUNT 模式下启动实例。

```
RMAN> STARTUP NOMOUNT  
      > PFILE='?/oradata/test/initiorcl.ora';
```

将数据库还原到新主机

9. 创建 RUN 块以执行下列任务：

- 还原控制文件
- 装载数据库

10. 创建 RMAN 恢复脚本以还原和恢复数据库。

11. 执行 RMAN 脚本。

12. 使用 RESETLOGS 选项打开数据库。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

将数据库还原到新主机（续）

9. 创建 RUN 块以便从自动备份中还原控制文件并装载数据库，如下例所示：

```
RUN
{
  RESTORE CONTROLFILE FROM AUTOBACKUP;
  ALTER DATABASE MOUNT;
}
```

10. 在新主机上查询 V\$DATAFILE 以确定数据库文件名（记录在控制文件中）。创建 RMAN 恢复脚本以还原和恢复数据库，根据实际情况可包括以下步骤：

- a. 对于还原目标位置不同于原始主机上的位置的每个数据文件，使用 SET NEWNAME 命令指定它在新主机上的路径。
- b. 使用 SQL ALTER DATABASE RENAME FILE 命令指定联机重做日志文件的路径。
- c. 使用 SET UNTIL 命令将恢复限制到归档重做日志文件的结尾。
- d. 请包括 SWITCH 命令，以便控制文件将新路径名识别为数据文件的正确名称。

将数据库还原到新主机（续）

以下是一个恢复脚本示例：

```

RUN
{
SET NEWNAME FOR DATAFILE 1 TO '?/oradata/test/system01.dbf';
SET NEWNAME FOR DATAFILE 2 TO '?/oradata/test/undotbs01.dbf';
SET NEWNAME FOR DATAFILE 3 TO '?/oradata/test/sysaux.dbf';
SET NEWNAME FOR DATAFILE 4 TO '?/oradata/test/users01.dbf';
SET NEWNAME FOR DATAFILE 5 TO '?/oradata/test/example01.dbf';
SQL "ALTER DATABASE RENAME FILE
    '/u01/app/oracle/oradata/orcl/redo01.log'
    TO '?/oradata/test/redo01.log' ";
SQL "ALTER DATABASE RENAME FILE
    '/u01/app/oracle/oradata/orcl/redo02.log'
    TO '?/oradata/test/redo02.log' ";
SQL "ALTER DATABASE RENAME FILE
    '/u01/app/oracle/oradata/orcl/redo03.log'
    TO '?/oradata/test/redo03.log' ";
SET UNTIL SCN 4545727;
RESTORE DATABASE;
SWITCH DATAFILE ALL;
RECOVER DATABASE;
}

```

11. 执行恢复脚本。

12. 使用 RESETLOGS 选项打开数据库：

```

RMAN> ALTER DATABASE OPEN RESETLOGS;

```

完成测试后，可关闭测试数据库实例并删除测试数据库及其所有文件。

执行灾难恢复

- 灾难意味着丢失了整个目标数据库、恢复目录数据库、所有当前控制文件、所有联机重做日志文件和所有参数文件。
- 灾难恢复包括还原和恢复目标数据库。
- 备份集的最低要求：
 - 数据文件的备份
 - 相应的归档重做日志文件
 - 至少一个控制文件自动备份

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行灾难恢复

灾难恢复包括在丢失了整个目标数据库、所有当前控制文件、所有联机重做日志文件、所有参数文件和恢复目录数据库（如果有）后还原和恢复目标数据库。

要执行灾难恢复，至少需要以下备份：

- 数据文件的备份
- 在备份之后生成的相应归档重做日志
- 至少一个控制文件自动备份

注：有关 Oracle Data Guard 如何提供完整灾难保护的信息，请参阅《Oracle Data Guard Concepts and Administration》手册。

执行灾难恢复

基本过程：

- 还原服务器参数文件的自动备份。
- 启动目标数据库实例。
- 从自动备份还原控制文件。
- 装载数据库。
- 还原数据文件。
- 恢复数据文件。
- 使用 RESETLOGS 选项打开数据库。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行灾难恢复（续）

本幻灯片概述了执行灾难恢复的基本过程。装载数据库后，请按照相应的步骤使用备份控制文件执行恢复。

测验

如果没有丢失任何数据文件，则恢复备份控制文件时，为什么需要使用 RECOVER 命令？

1. 通过从数据文件进行重新同步，前滚对控制文件的更改
2. 通过从重做日志应用重做，前滚对控制文件的更改
3. 通过使用 RMAN 目录，前滚对控制文件的更改

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2

测验

使用 RESTORE 命令，可从备份还原数据库文件，但不可从重做日志应用重做。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1

小结

在本课中，您应该已经学会如何使用 RMAN 执行以下任务：

- 在丢失关键或非关键数据文件后执行完全恢复
- 使用增量更新的备份进行恢复
- 切换到映像副本进行快速恢复
- 将数据库还原到新主机上
- 使用备份控制文件进行恢复

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 7 概览：使用 RMAN 执行恢复

本练习包含以下主题：

- 恢复映像副本
- 执行快速恢复

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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

8 监视和优化 RMAN

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

学完本课后，应能完成以下工作：

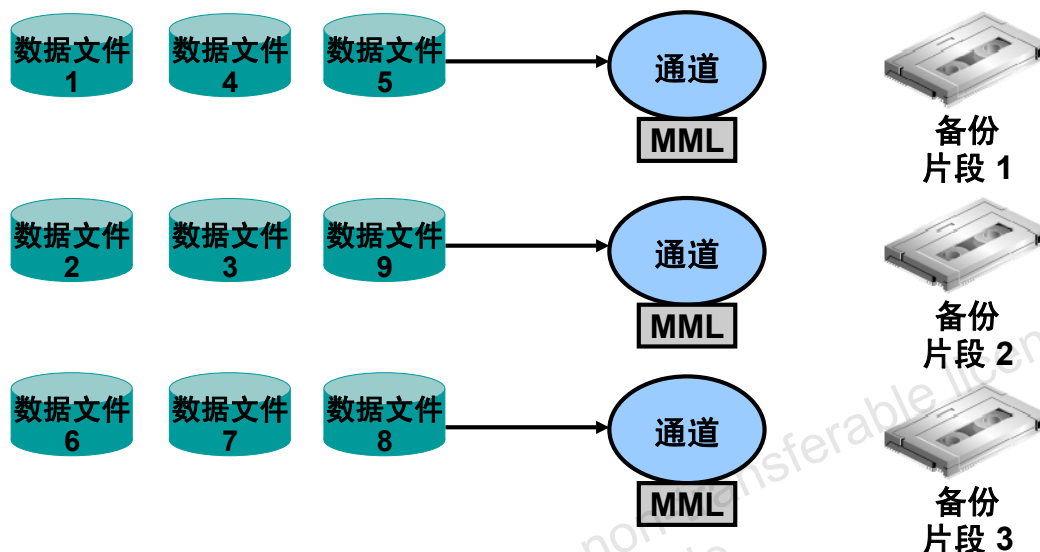
- 监视 RMAN 作业进度
- 针对异步 I/O 对 RMAN 进行相应配置
- 配置 RMAN 多路复用以保持磁带机流高效运行
- 评估备份速度与恢复速度之间的平衡
- 说明下列参数对 RMAN 性能的影响：MAXPIECESIZE、FILESERSET、MAXOPENFILES
- 说明 RMAN BACKUP DURATION 选项如何加快或降低备份的执行速度（释放资源以进行其它处理）

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

并行备份集

为提高性能，分配多个通道并将文件分配给特定通道。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

并行备份集

可通过将 CONFIGURE 命令的 PARALLELISM 选项设置为大于 1 的值，或通过手动分配多个通道来配置并行备份。RMAN 并行执行其操作，然后并行写入多个备份集。多个服务器会话承担备份指定文件的工作。

示例

```
RMAN> RUN {
2>   ALLOCATE CHANNEL c1 DEVICE TYPE sbt;
3>   ALLOCATE CHANNEL c2 DEVICE TYPE sbt;
4>   ALLOCATE CHANNEL c3 DEVICE TYPE sbt;
5>   BACKUP
6>   INCREMENTAL LEVEL = 0
7>   (DATAFILE 1,4,5 CHANNEL c1)
8>   (DATAFILE 2,3,9 CHANNEL c2)
9>   (DATAFILE 6,7,8 CHANNEL c3);
10>  SQL 'ALTER SYSTEM ARCHIVE LOG CURRENT';
11> }
```

备份数据文件时，可以通过其路径名或文件编号指定要备份的文件。例如，以下两个命令执行相同的操作：

```
BACKUP DEVICE TYPE sbt DATAFILE '/home/oracle/system01.dbf';
BACKUP DEVICE TYPE sbt DATAFILE 1;
```

并行备份集（续）

创建多个备份集并分配多个通道时，RMAN 自动并行执行其操作，然后并行写入多个备份集。分配的服务器会话共同承担备份指定数据文件、控制文件和归档重做日志的工作。不能跨多个通道条带化一个备份集。

可通过以下方法实现备份集的并行：

- 将 PARALLELISM 的值配置为大于 1 或分配多个通道。
- 指定要备份的多个文件。

示例

- 有九个需要备份的文件（数据文件 1 到 9）。
- 将多个数据文件分配给一个备份集，以使每个集所包含的要备份的数据块数目大致相同（为提高效率）。
 - 数据文件 1、4 和 5 被分配到备份集 1。
 - 数据文件 2、3 和 9 被分配到备份集 2。
 - 数据文件 6、7 和 8 被分配到备份集 3。

注：还可以使用 FILESPERSET 参数来限制备份集中包含的数据文件的数目。

监视 RMAN 会话

- 通过查询 V\$SESSION 和 V\$PROCESS 可确定服务器会话和 RMAN 通道之间的关系。
- 如果正在监视多个会话，请使用 SET COMMAND ID 命令在备份期间将某个进程与某个通道相关联。

```
SQL> COLUMN CLIENT_INFO FORMAT a30
SQL> COLUMN SID FORMAT 999
SQL> COLUMN SPID FORMAT 9999
SQL> SELECT s.sid, p.spid, s.client_info
  2 FROM v$process p, v$session s
  3 WHERE p.addr = s.paddr
  4 AND CLIENT_INFO LIKE 'rman%';
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

监视 RMAN 会话

要确定哪些服务器会话与哪些 RMAN 通道相对应，可以查询 V\$SESSION 和 V\$PROCESS。V\$PROCESS 的 SPID 列确定了进程或线程的操作系统 ID 号。在 UNIX 中，SPID 列显示进程 ID。在 Windows 中，SPID 列显示线程 ID。有两种基本方式可获取此信息，具体取决于是否有多个 RMAN 会话同时处于活动状态。当只有一个 RMAN 会话处于活动状态时，请在 RMAN 作业正在运行时对目标数据库执行下列查询：

```
SQL> COLUMN CLIENT_INFO FORMAT a30
SQL> COLUMN SID FORMAT 999
SQL> COLUMN SPID FORMAT 9999
SQL> SELECT s.sid, p.spid, s.client_info
  2 FROM v$process p, v$session s
  3 WHERE p.addr = s.paddr
  4 AND CLIENT_INFO LIKE 'rman%';
SID SPID CLIENT_INFO
-----
15 2714 rman channel=ORA_SBT_TAPE_1
13 2715 rman channel=ORA_SBT_TAPE_2
```

监视 RMAN 会话（续）

当有多个 RMAN 会话正在运行时，最好使用 SET COMMAND ID 命令在备份期间将某个进程与某个通道相关联，如下所示：

1. 在每个会话中，将命令 ID 设置为不同的值，然后备份所需的对象。例如，在会话 1 中输入下列内容：

```
RUN
{
  SET COMMAND ID TO 'sess1';
  BACKUP DATABASE;
}
```

在会话 2 中运行的作业中将命令 ID 设置为一个字符串，如 sess2：

```
RUN
{
  SET COMMAND ID TO 'sess2';
  BACKUP DATABASE;
}
```

2. 启动 SQL*Plus 会话，然后在执行 RMAN 作业时查询联接的 V\$SESSION 和 V\$PROCESS 视图。例如，输入：

```
SELECT SID, SPID, CLIENT_INFO
FROM V$PROCESS p, V$SESSION s
WHERE p.ADDR = s.PADDR
AND CLIENT_INFO LIKE '%id=sess%';
```

如果在 RMAN 作业中运行 SET COMMAND ID 命令，则 CLIENT_INFO 列会以下列格式显示：

id=command_id,rman channel=channel_id

例如，下面显示了一个示例输出：

SID	SPID	CLIENT_INFO
11	8358	id=sess1
15	8638	id=sess2
14	8374	id=sess1,rman channel=c1
9	8642	id=sess2,rman channel=c1

监视 RMAN 作业进度

通过查询 V\$SESSION_LONGOPS 监视备份和还原操作的进度。

```
SQL> SELECT OPNAME, CONTEXT, SOFAR, TOTALWORK,
2  ROUND(SOFAR/TOTALWORK*100,2) "%_COMPLETE"
3  FROM V$SESSION_LONGOPS
4  WHERE OPNAME LIKE 'RMAN%'
5  AND OPNAME NOT LIKE '%aggregate%'
6  AND TOTALWORK != 0
7  AND SOFAR <> TOTALWORK;
```

SID	SERIAL#	CONTEXT	SOFAR	TOTALWORK	%_COMPLETE
13	75	1	9470	15360	61.65
12	81	1	15871	28160	56.36

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

监视 RMAN 作业进度

通过查询 V\$SESSION_LONGOPS 视图可监视备份、复制和还原的进度。RMAN 使用 V\$SESSION_LONGOPS 中的详细信息行和聚集行。详细信息行说明了在一个作业步骤中处理的文件。聚集行说明了在 RMAN 命令的所有作业步骤中处理的文件。作业步骤是指创建或还原一个备份集或一个数据文件副本。详细信息行使用备份步骤期间读写的每一个缓冲区来进行更新，因此它们的更新粒度比较小。聚集行在每个作业步骤完成后才进行更新，因此它们的更新粒度比较大。

注：将 STATISTICS_LEVEL 参数设置为 TYPICAL（默认值）或 ALL 可填充 V\$SESSION_LONGOPS 视图。

V\$SESSION_LONGOPS 中与 RMAN 相关的列包括：

- **OPNAME:** 关于该行的文字说明。详细信息行包括 RMAN:datafile copy、RMAN:full datafile backup 和 RMAN:full datafile restore。
- **CONTEXT:** 对于备份输出行，此列的值为 2。对于除代理副本（不更新此列）外的其它所有行，值为 1。
- **SOFAR:** 对于映像副本，是指已读取的块数；对于备份输入行，是指已从所备份的文件中读取的块数；对于备份输出行，是指已写入备份片段的块数；对于还原，是指已处理到通过一个作业步骤还原的文件的块数；对于代理副本，是指已复制的文件数。

监视 RMAN 作业进度（续）

- **TOTALWORK:** 对于映像副本，是指文件中的总块数；对于备份输入行，是指要从通过此作业步骤处理的所有文件中读取的总块数；对于备份输出行，值为 0，因为 RMAN 不知道要写入到任何备份片段的块数；对于还原，是指此作业步骤中还原的所有文件中的总块数；对于代理副本，是指在此作业步骤中要复制的文件总数。

解释 RMAN 消息输出

可在以下内容中找到 RMAN 故障诊断信息：

- RMAN 命令输出
- RMAN 跟踪文件
- 预警日志
- Oracle Server 跟踪文件
- sbtio.log 文件

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

解释 RMAN 消息输出

RMAN 命令输出包含与 RMAN 作业有关的操作，以及 RMAN、服务器和介质供应商生成的错误消息。RMAN 错误消息有 RMAN-nnnn 前缀。输出会显示到终端（标准输出），但是通过定义 LOG 选项或者通过 shell 重定向可将其写入到文件。

RMAN 跟踪文件包含 DEBUG 输出并且只在使用 TRACE 命令选项时才使用。

预警日志包含按时间顺序列出的错误、非默认初始化参数设置以及管理操作的日志。由于它记录了已改写的控制文件记录的值，因此在没有恢复目录的情况下操作时，该日志可用于 RMAN 维护。

Oracle 跟踪文件包含 Oracle Server 进程生成的详细输出。当出现 ORA-600 或 ORA-3113（跟在 ORA-7445 后）错误消息时，每当 RMAN 不能分配通道，并且不能加载介质管理库时，就会创建这个文件。可在 USER_DUMP_DEST 中找到该文件。

sbtio.log 文件包含由介质管理软件写入的、特定于供应商的信息，可在 USER_DUMP_DEST 中找到此文件。请注意，此日志不包含 Oracle Server 或 RMAN 错误。

使用 DEBUG 选项

- DEBUG 选项用于：
 - 查看生成的 PL/SQL
 - 准确判断 RMAN 命令在何处挂起或出错
- DEBUG 选项是在 RMAN 提示符下或者在 run 块内指定的。
- DEBUG 选项会创建大量输出，因此要将输出重定向到跟踪文件：

```
$ rman target / catalog rman/rman debug trace trace.log
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 DEBUG 选项

DEBUG 选项显示在 RMAN 编译期间执行的全部 SQL 语句以及执行结果。另外，还显示恢复目录 PL/SQL 程序包生成的所有信息。以下示例中，在备份数据文件 3 期间会写入 DEBUG 输出，但是在备份数据文件 4 期间则不会：

```
RMAN> run {
    debug on;
    allocate channel c1 type disk;
    backup datafile 3;
    debug off;
    backup datafile 4; }
```

请记住，DEBUG 输出可能非常大，所以请确保有足够的磁盘空间可供跟踪文件使用。

这个示例备份会话中没有生成任何错误，只创建了一个大约 0.5 兆字节大小的跟踪文件：

```
$ rman target / catalog rman/rman debug trace sample.log
RMAN> backup database;
RMAN> host "ls -l sample.log";
-rw-r--r--  1 user02  dba          576270 Apr  6 10:38 sample.log
host command complete
```


解释 RMAN 错误堆栈

- 从下到上读取堆栈。读取堆栈。
- 查找 Additional information。
- RMAN-03009 指示失败的命令。

```

RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03009: failure of backup command on c1 channel at
              09/04/2001 13:18:19
ORA-19506: failed to create sequential file,
              name="07d36ecp_1_1", parms=""
ORA-27007: failed to open file
SVR4 Error: 2: No such file or directory
Additional information: 7005
Additional information: 1
ORA-19511: Error from media manager layer,error text:
  
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

解释 RMAN 错误堆栈

由于 RMAN 记录的数据量很大，您可能发现很难在 RMAN 错误堆栈中找到有用的消息。请注意以下提示和建议：

- 由于错误堆栈中的许多消息对执行故障诊断没有什么意义，所以请尽量找到一两条重要的错误。
- 找到显示有 Additional information（后面跟有一个整数）的行。此行指示介质管理错误。后面跟着的整数是指错误消息文本中解释的代码。
- 从下到上读取堆栈。从下到上读取消息，因为这是 RMAN 发布消息的顺序。堆栈中显示的最后一条或两条错误通常是参考性消息。
- 查找紧跟着标帜的 RMAN-03002 或 RMAN-03009 消息。RMAN-03009 与 RMAN-03002 相同，但包括通道 ID。如果故障与 RMAN 命令有关，则这些消息指示哪个命令执行失败。语法错误会生成 RMAN-00558 错误。

优化 RMAN

- RMAN BACKUP 和 RESTORE 操作执行下列任务：
 - 读取或写入数据。
 - 通过复制和验证块来处理数据。
- 对于任何特定进程，这些任务中最慢的任务被称为瓶颈。
- 优化 RMAN 需要找出并解决瓶颈。
- 可以平衡备份与恢复操作的性能来满足您的需要。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

优化 RMAN

RMAN 的备份和还原操作会执行下列不同任务：

- 读取或写入输入数据
- 通过验证块并将块从输入缓冲区复制到输出缓冲区来处理数据

这些任务中最慢的任务被称为“瓶颈”。RMAN 优化需要找出瓶颈，然后使用 RMAN 命令、初始化参数设置或调整物理介质使瓶颈变得更高效。优化 RMAN 的关键是了解输入/输出 (I/O)。RMAN 备份和还原作业使用两种类型的 I/O 缓冲区：磁盘和三级存储（通常为磁带）。执行备份时，RMAN 通过使用磁盘缓冲区读取输入文件，通过使用磁盘或磁带缓冲区写入输出备份文件。执行还原操作时，RMAN 会撤消这些角色。I/O 可以是同步的，也可以是异步的。同步设备一次只执行一个 I/O 任务。因此，能够很容易地确定备份作业所需的时间。与同步 I/O (SIO) 相反，异步 I/O (AIO) 一次可以执行多个任务。要有效地优化 RMAN，必须全面了解同步 I/O 与异步 I/O、磁盘缓冲区与磁带缓冲区以及通道体系结构的概念。了解这些概念之后，就可以使用固定的视图来监视瓶颈。

可以利用某些备份和恢复功能来平衡备份操作与恢复操作的性能。例如，如果需要加快恢复速度，则可以定期执行映像副本恢复。这会占用更多资源来准备恢复，但会减少执行恢复所需的时间。

RMAN 多路复用

- 对于读取操作：

多路复用级别	分配规则
级别 ≤ 4	分配 1 MB 缓冲区，这样所有输入文件的缓冲区总大小为 16 MB。
$4 < \text{级别} \leq 8$	分配 512 KB，这样所有文件的缓冲区总大小低于 16 MB。
级别 > 8	RMAN 在每个通道中为每个文件分配四个 128 KB 磁盘缓冲区，这样在每个通道中为每个文件分配的磁盘缓冲区的总大小为 512 KB。

- 对于写入操作，每个通道分配四个输出缓冲区，每个 1 MB。

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

RMAN 多路复用

RMAN 对 I/O 使用两种不同类型的缓冲区：磁盘和磁带。RMAN 多路复用确定了 RMAN 如何分配磁盘缓冲区。“RMAN 多路复用”是指在备份中同时读取然后写入到同一备份片段的文件数。多路复用的程度取决于 BACKUP 命令的 FILESPERSET 参数以及 CONFIGURE CHANNEL 命令或 ALLOCATE CHANNEL 命令的 MAXOPENFILES 参数。注：RMAN 多路复用是在通道级别设置的。

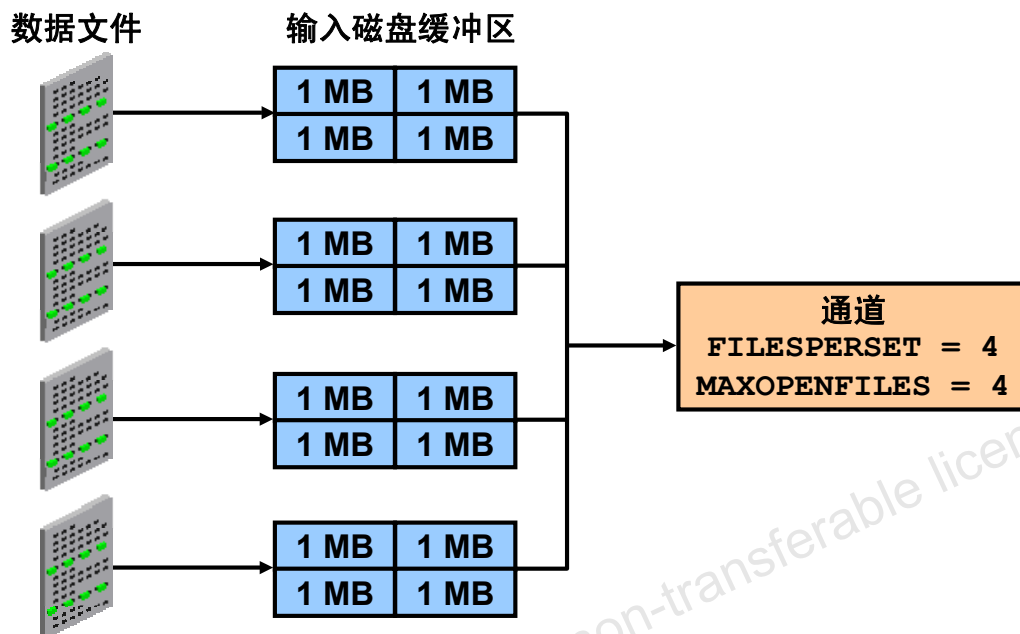
例如，假定使用一个通道备份两个数据文件。将 FILESPERSET 设置为 3，并将 MAXOPENFILES 设置为 8。在本例中，每个备份集中的文件数为 2（FILESPERSET 与每个通道读取的文件数之中的较小值），并且多路复用级别为 2（MAXOPENFILES 与每个备份集中的文件数之中的较小值）。当 RMAN 从磁盘备份时，它使用幻灯片中的表中介绍的算法。

对于写入操作，每个通道分配四个输出缓冲区，每个缓冲区的大小为 1 MB。

除非 DBWR_IO_SLAVES 设置为非零值，否则这些缓冲区将从 PGA 进行分配。

注：为了获得最佳恢复性能，请勿将 FILESPERSET 设置为大于 8 的值。

分配磁盘缓冲区：示例



版权所有 © 2011, Oracle. 保留所有权利。

分配磁盘缓冲区：示例

在本幻灯片示例中，使用一个通道备份四个数据文件。MAXOPENFILES 设置为 4 且 FILESPERSET 设置为 4。本示例中多路复用级别为 4。每个数据文件的缓冲区总大小为 4 MB。要计算备份集中分配的缓冲区总大小，请用每个数据文件的总字节数乘以通道当前访问的数据文件数，然后用所得值乘以通道数。

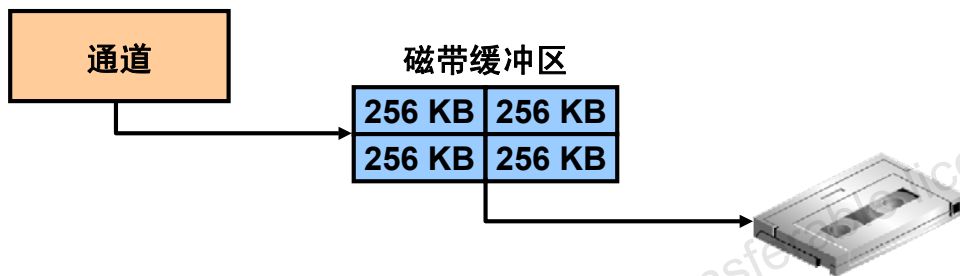
假定使用一个通道备份四个数据文件，并且使用幻灯片中显示的设置。这种情况下，按如下方式相乘可得到为备份分配的缓冲区总大小：

每个数据文件 4 MB × 1 个通道 × 每个通道 4 个数据文件 = 16 MB

请设置 MAXOPENFILES 参数，以使同时读取的文件数正好可以完全使用输出设备。当输出设备为磁带时，这个注意事项很重要。

分配磁带缓冲区

- 从 SGA（大型池）分配，BACKUP_TAPE_IO_SLAVES 为 TRUE。
- 从 PGA 分配，BACKUP_TAPE_IO_SLAVES 为 FALSE。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

分配磁带缓冲区

如果备份到磁带设备，Oracle Server 会在每个通道中为磁带写进程（如果执行还原则为读进程）分配四个缓冲区。Oracle Server 仅在通道为系统备份到磁带 (SBT) 通道时才分配这些缓冲区。通常，每个磁带缓冲区为 256 KB。要计算执行备份或还原期间使用的缓冲区总大小，请用缓冲区大小乘以四，然后用所得值乘以通道数。

如幻灯片中的示例所示，假定使用一个磁带通道，每个缓冲区为 256 KB。这种情况下，备份期间所用的缓冲区总大小如下：

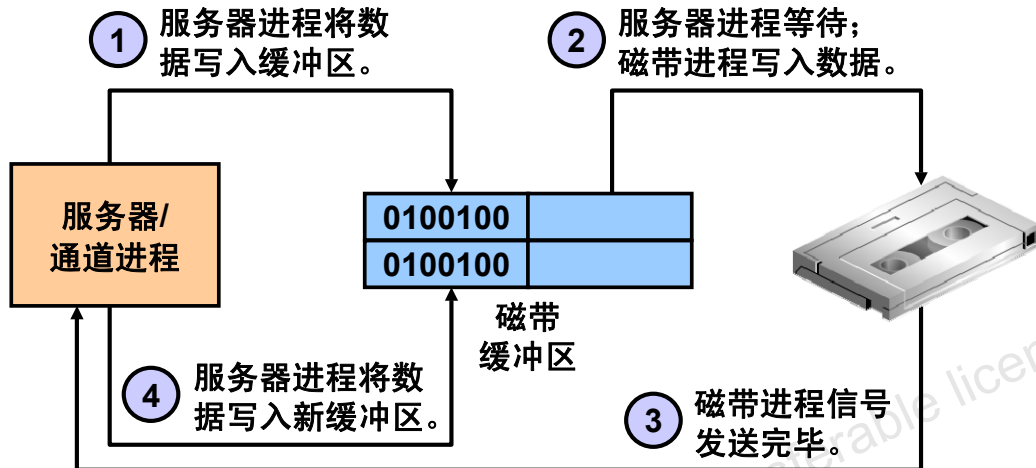
$$\text{每个缓冲区 } 256 \text{ KB} \times \text{每个通道 } 4 \text{ 个缓冲区} \times 1 \text{ 个通道} = 1,024 \text{ KB}$$

RMAN 会在系统全局区 (SGA) 或程序全局区 (PGA) 分配磁带缓冲区，具体取决于是否使用 I/O 从属进程。如果 BACKUP_TAPE_IO_SLAVES 初始化参数设置为 TRUE，则当设置了 LARGE_POOL_SIZE 初始化参数时，RMAN 会从共享池或大型池分配磁带缓冲区。如果将该参数设置为 FALSE，RMAN 会从 PGA 分配缓冲区。如果使用 I/O 从属进程，请设置 LARGE_POOL_SIZE 初始化参数，以便留出一些 SGA 内存专门用来支持这些大型内存分配。这样，RMAN I/O 缓冲区就不会与库高速缓存争用共享池内存。

Oracle 建议您将 BACKUP_TAPE_IO_SLAVES 初始化参数设置为 TRUE。大多数情况下，这在备份到磁带时可以实现最佳性能。执行双向备份时也需要该设置。“使用 RMAN 创建备份”一课中介绍了双向备份。

比较同步 I/O 和异步 I/O

同步 I/O



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

比较同步 I/O 和异步 I/O

当 RMAN 读取或写入数据时，I/O 是同步或异步的。当 I/O 为同步的时，服务器进程一次只能执行一个任务。当 I/O 为异步的时，服务器进程可以先开始一个 I/O，然后在等待该 I/O 完成期间执行其它任务。也可以在等待第一个 I/O 完成之前开始多个 I/O 操作。

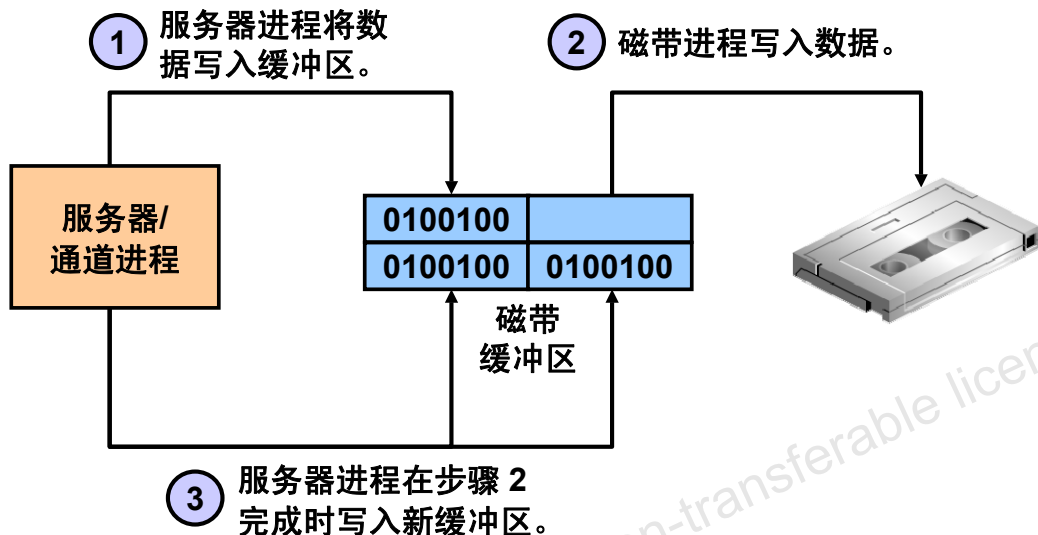
您可以设置决定 I/O 类型的初始化参数。如果将 BACKUP_TAPE_IO_SLAVES 设置为 TRUE，则磁带 I/O 是异步的。否则，I/O 是同步的。

幻灯片中的示例显示了在备份到磁带时使用的同步 I/O。同步传输时会执行以下步骤：

1. 服务器进程将块写入磁带缓冲区。
2. 磁带进程将数据写入磁带。当介质管理器将数据从 Oracle 缓冲区复制到介质管理器的内部缓冲区时，服务器进程处于空闲状态。
3. 磁带进程中继到已完成写入操作的服务器进程。
4. 服务器进程可以启动新任务。

比较同步 I/O 和异步 I/O

异步 I/O



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

比较同步 I/O 和异步 I/O（续）

许多操作系统都支持本机异步 I/O，Oracle 可以利用这一功能（只要该功能可用）。如果平台支持，建议您始终将 `BACKUP_TAPE_IO_SLAVES` 设置为 `TRUE`。在不支持本机异步 I/O 的操作系统上，Oracle 通过使用专门代表其它进程执行 I/O 的特殊 I/O 从属进程可模拟这一功能。通过将 `DBWR_IO_SLAVES` 参数设置为非零值可控制磁盘 I/O 从属进程。Oracle 为任何非零值 `DBWR_IO_SLAVES` 分配四个备份磁盘 I/O 从属进程。

幻灯片中的示例显示了在备份到磁带时使用的异步 I/O。下面列出了在异步交换时执行的详细步骤：

1. 服务器进程将块写入磁带缓冲区。
2. 磁带进程将数据写入磁带。在磁带进程写入数据时，其它服务器进程可自由处理更多的输入块并填充更多的输出缓冲区。
3. 在初始磁带进程对磁带执行写入时，衍生的服务器进程对磁带缓冲区执行写入。

监视 RMAN 作业的性能

- 以下视图可用于监视备份和还原的性能：
 - V\$BACKUP_SYNC_IO
 - V\$BACKUP_ASYNC_IO
- 对于备份或还原存在以下各行：
 - 每个数据文件对应一行
 - 一个聚集数据文件行
 - 每个备份片段对应一行
- I/O 是否为同步的取决于控制进程如何查看它。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

监视 RMAN 作业的性能

最大备份速度受可用硬件限制。备份速度不可能快于汇总的磁带带宽，除非数据文件中存在许多不需要备份的空块。

备份系统组件中的一个将成为瓶颈，这取决于磁盘、磁带机和任何其它传输组件（如网络）的相对速度。例如，如果瓶颈是磁带机，并且磁带是流式，则备份速度不可能更快。

注：如果使用同步 I/O 并且已将 BACKUP_DISK_IO_SLAVES 初始化参数设置为 TRUE，则 I/O 将显示在 V\$BACKUP_ASYNC_IO 中。

异步 I/O 瓶颈

- 使用 V\$BACKUP_ASYNC_IO 监视异步 I/O。
- LONG_WAITS 与 IO_COUNT 之比最大的文件可能是瓶颈。
 - IO_COUNT: 对文件执行的 I/O 数
 - LONG_WAITS: “备份/还原进程指示操作系统必须等待 I/O 完成” 这一情况的发生次数
- 等待次数应为零才能避免瓶颈。
 - SHORT_WAIT_TIME_TOTAL
 - LONG_WAIT_TIME_TOTAL

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

异步 I/O 瓶颈

可使用 V\$BACKUP_ASYNC_IO 监视异步 I/O。LONG_WAITS 列显示 “备份或还原进程指示操作系统必须等待 I/O 完成” 这一情况的发生次数。SHORT_WAITS 列显示备份/还原进程执行操作系统调用以在不中断模式下轮询 I/O 是否完成的次数。在某些平台上，异步 I/O 实施可能会致使在对 I/O 执行不中断轮询时调用进程等待 I/O 完成。

找出瓶颈的最简单方法是通过查询 V\$BACKUP_ASYNC_IO 来查找 LONG_WAITS 与 IO_COUNT 之比最大的数据文件。

同步 I/O 瓶颈

- 同步 I/O 被视为一个瓶颈。
- 查询 V\$BACKUP_SYNC_IO 中的 DISCRETE_BYTES_PER_SECOND 列来查看 I/O 比率。
 - 将这个比率与设备的最大比率进行比较。
 - 如果该比率低于设备指定的值，则可进行优化。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

同步 I/O 瓶颈

使用同步 I/O 时，可以很容易地确定备份作业所需的时间，因为设备一次只执行一个 I/O 任务。Oracle I/O 使用轮询机制而不是中断机制来确定每个 I/O 请求完成的时间。由于操作系统不会在 I/O 完成时立即通知备份或还原进程，所以您不能确定每个 I/O 的持续时间。

使用 V\$BACKUP_SYNC_IO 可确定备份或还原瓶颈的根源，还可确定备份作业的进度。V\$BACKUP_SYNC_IO 包含 I/O 与执行备份的进程（在某些平台上为线程）同步时的行。

通道优化

使用 `CONFIGURE CHANNEL` 和 `ALLOCATE CHANNEL` 命令可执行以下操作：

- 限制备份片段的大小
- 防止 RMAN 占用太多的磁盘带宽
- 确定每个通道的多路复用级别
- 配置多个磁盘，以便将 I/O 活动分布在多个设备上
- 在 SBT 设备上配置多个通道，以便将不同的数据文件分配到各个通道

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

通道优化

可以使用 `CONFIGURE CHANNEL` 和 `ALLOCATE CHANNEL` 命令设置各种通道限制参数，这些参数将应用于由所分配的服务器会话执行的操作。

`MAXPIECESIZE` 参数指定备份片段的最大大小。使用这个参数可指示 RMAN 在一个备份集中创建多个备份片段。RMAN 创建的每个备份片段的大小不会超过此参数指定的值。

`RATE` 参数指定 RMAN 每秒在每个通道上读取的字节数。这个参数可用于防止 RMAN 占用过多的磁盘带宽和降低联机事务处理 (OLTP) 性能。例如，如果每个磁盘驱动器每秒提交 3 MB，而设置了 `RATE=1500K`，则仍有一些磁盘带宽可供联机系统使用。

`MAXOPENFILES` 参数决定了备份或复制在给定时间可打开的输入文件的最大数。如果未手动设置该参数，则默认值为 8。RMAN 多路复用级别在某些程度上由 `MAXOPENFILES` 决定。多路复用级别反过来又决定了 RMAN 如何分配磁盘缓冲区。多路复用是指同时读取然后写入同一备份片段的输入文件数。

通道优化（续）

如果为 SBT 设备配置了多个通道，则可以专门将数据文件分布在这些通道上。下面是一个示例：

```
RUN
{
    ALLOCATE CHANNEL c1 DEVICE TYPE sbt;
    ALLOCATE CHANNEL c2 DEVICE TYPE sbt;
    ALLOCATE CHANNEL c3 DEVICE TYPE sbt;
    BACKUP (DATAFILE 1,2,5 CHANNEL c1)
           (DATAFILE 4,6 CHANNEL c2)
           (DATAFILE 3,7,8 CHANNEL c3);
    BACKUP DATABASE NOT BACKED UP;
}
```

优化 BACKUP 命令

- MAXPIECESIZE 限制了每个备份片段的大小。
- FILESPERSET 可防止 RMAN 一次从太多磁盘读取数据。
- 如果没有将 MAXOPENFILES 设置得足够大，可能会禁止流式传输到磁带。
- BACKUP DURATION 可降低备份操作给系统带来的负载量。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

优化 BACKUP 命令

MAXPIECESIZE 参数指定在通道中创建的每个备份片段的最大大小。

FILESPERSET 参数指定在一个备份集中最多放多少个文件。如果只分配一个通道，则可以使用这个参数使 RMAN 创建多个备份集。例如，如果有 50 个输入数据文件和两个通道，则可以设置 FILESPERSET=5 以创建 10 个备份集。此策略可防止将一个备份集拆分到多个磁带上。

MAXOPENFILES 参数设置取决于磁盘子系统特性。如果使用 ASM，则可将其设置为 1 或 2。其他情况下，如果数据未条带化，则可能需要将其设置为更大的值。为提高性能，请增加每个备份集的文件数，或增大此参数。如果未使用 ASM 或任何种类的条带化，请尝试增加 MAXOPENFILES。

可以通过不同方式使用 BACKUP 命令的 BACKUP DURATION 选项。如果指定的持续时间比完成备份所需的时间短，则可以使用此选项将备份活动保留在特定时间窗口内。在特定情况下，未完成的部分备份不会丢失。

此外，该选项具有两个修饰符：

- **MINIMIZE TIME:** 备份尽可能快地运行。
- **MINIMIZE LOAD:** 备份尝试使用时间窗口中可用的全部时间量。这可以减少系统负载。

优化 RMAN 备份性能

要优化 RMAN 备份性能，请执行以下步骤：

1. 从已配置和分配的通道中删除 RATE 设置。
2. 如果使用同步磁盘 I/O，请设置 DBWR_IO_SLAVES。
3. 设置 LARGE_POOL_SIZE。
4. 优化 RMAN 磁带流性能瓶颈问题。
5. 通过查询 V\$ 视图找出瓶颈。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

优化 RMAN 备份性能

要得到最佳备份性能，请遵循以下步骤进行操作：

1. 从已配置和分配的通道中删除 RATE 设置。RATE 参数用于设置 RMAN 每秒在通道上读取的最大字节数（默认值）、最大千字节数 (K)、最大兆字节数 (M) 或最大千兆字节数 (G)。它设置读取字节的上限，以便 RMAN 不占用过多的磁盘带宽，因而不会降低性能。如果备份未流式传输到磁带，请确保没有在 ALLOCATE CHANNEL 或 CONFIGURE CHANNEL 命令中设置 RATE 参数。
2. 如果使用的是同步磁盘 I/O，请设置 DBWR_IO_SLAVES。如果磁盘不支持异步 I/O，请尝试将 DBWR_IO_SLAVES 初始化参数设置为一个非零值。将 DBWR_IO_SLAVES 设置为任何非零值都会导致在备份和还原中使用固定数量（四个）的磁盘 I/O 从属进程，因此会模拟异步 I/O。如果使用了 I/O 从属进程，则从 SGA 获得 I/O 缓冲区。如果配置了大型池，则使用大型池。否则，使用共享池。
注：通过设置 DBWR_IO_SLAVES，数据库写进程也会使用从属进程。您可能需要增大 PROCESSES 初始化参数的值。
3. 按下页所述设置 LARGE_POOL_SIZE 的值。
4. 按本课后部分中的内容优化 RMAN 磁带流性能瓶颈问题。
5. 按本课前面部分中的内容使用 V\$ 视图。

设置 LARGE_POOL_SIZE

- 如果未设置 LARGE_POOL_SIZE, Oracle Server 会尝试从共享池获取内存。
- 如果所设置的 LARGE_POOL_SIZE 值不够大, 服务器不会从共享池分配缓冲区。
- 如果服务器得不到足够的内存, 则会从本地进程内存分配缓冲区。
- Oracle Server 将消息写入预警日志, 指示此备份使用了同步 I/O。

```
ksfqxcrc: failure to allocate shared memory means sync
I/O will be used whenever async I/O to file not
supported natively
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

设置 LARGE_POOL_SIZE

从共享池请求的连续内存分配量很小, 通常不到 5 KB。如果请求的连续内存分配量比较大, 则请求可能失败, 或者可能需要进行大量内存调整来释放所需的连续内存量。大型池能够满足这种内存请求。大型池没有最近最少使用列表, 因此 Oracle 不会尝试从大型池中清除内存。

使用 LARGE_POOL_SIZE 初始化参数可配置大型池。通过查询 V\$SGASTAT.POOL 可查看对象内存位于哪个池 (共享池或大型池) 中。建议的 LARGE_POOL_SIZE 值是按以下公式计算的:

$$\#_of_allocated_channels * (16 \text{ MB} + (4 * size_of_tape_buffer))$$

对于备份到磁盘, 磁带缓冲区显然为 0, 因此请将 LARGE_POOL_SIZE 设置为 16 MB。对于磁带备份, 单个磁带缓冲区的大小由 RMAN 通道参数 BLKSIZE 定义, 其默认值为 256 KB。假定要备份到两个磁带机。如果磁带缓冲区大小为 256 KB, 请将 LARGE_POOL_SIZE 设置为 18 MB。如果将 BLKSIZE 增大至 512 KB, 请将 LARGE_POOL_SIZE 增大至 20 MB。

注: 仅当 DBWR_IO_SLAVES > 0 时, 才为磁盘缓冲区使用大型池, 仅当 BACKUP_TAPE_IO_SLAVES = TRUE 时, 才为磁带缓冲区使用大型池。如果您在使用自动共享内存管理功能, 则大型池大小会根据系统工作量自动调整。

优化 RMAN 磁带流性能瓶颈问题

- 使用 `BACKUP... VALIDATE` 可确定瓶颈是磁带流还是磁盘 I/O。
- 使用多路复用可改善磁盘瓶颈中的磁带流性能。
- 使用增量备份可改善磁带备份性能瓶颈。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

优化 RMAN 磁带流性能瓶颈问题

要找出并消除影响 RMAN 磁带备份性能的瓶颈，请执行以下操作：

- 使用 `BACKUP... VALIDATE` 可确定给定备份作业中的瓶颈是磁带流还是磁盘 I/O。将运行备份任务所需的时间与对同一任务运行 `BACKUP VALIDATE` 所需的时间进行比较。备份到磁带的 `BACKUP VALIDATE` 会执行与真正备份相同的磁盘读操作，但不执行磁带 I/O。如果磁带的 `BACKUP VALIDATE` 所需时间远远低于真正备份到磁带所需的时间，则到磁带的写入很可能就是瓶颈所在。
- 使用多路复用可改善磁盘瓶颈中的磁带流性能。某些情况下，当 RMAN 执行到磁带的备份时，向磁带机传送数据块的速度可能不足以支持磁带流。例如，在增量备份期间，作为同一策略的一部分 RMAN 只备份自上次数据文件备份以来更改过的块。如果未启用更改跟踪，RMAN 必须扫描整个数据文件来查找更改的块，然后在找到此类块后填充输出缓冲区。如果更改的块不多，RMAN 就不能足够快地填充输出缓冲区来维持磁带机流。此时，通过增大备份使用的多路复用程度可提高性能。这会加快 RMAN 填充磁带缓冲区的速率，可以足够快的速度向介质管理器传送缓冲区，从而维持磁带流。

优化 RMAN 磁带流性能瓶颈问题（续）

- 使用增量备份可改善磁带备份性能瓶颈。如果写入到磁带是执行备份时的瓶颈来源，可以考虑使用增量备份作为备份策略的一部分。1 级增量备份只会将数据文件中更改的块写入磁带，因此会减小写入磁带时的任何瓶颈对整个备份策略的影响。尤其是当磁带机未连接到运行备份数据库的节点本地时，采用增量备份的速度会更快。

测验

选择关于 RMAN 优化的正确语句：

1. 可通过将 CONFIGURE 命令的 PARALLELISM 选项设置为大于 1 的值，或通过手动分配多个通道来配置并行备份。
2. 可以跨多个通道条带化一个备份集来提高性能。
3. 每次提高备份操作的速度时，会自动提高还原和恢复操作的速度。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1

测验

您从不会遇到 RMAN 瓶颈，因为优化指导会自动将其修复。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2

小结

在本课中，您应该已经学会：

- 监视 RMAN 作业进度
- 针对异步 I/O 对 RMAN 进行相应配置
- 配置 RMAN 多路复用以保持磁带机流高效运行
- 评估备份速度与恢复速度之间的平衡
- 说明下列参数对 RMAN 性能的影响：MAXPIECESIZE、FILESERSET、MAXOPENFILES
- 说明 RMAN BACKUP DURATION 选项如何加快或降低备份的执行速度（释放资源以进行其它处理）

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 8 概览：监视和优化 RMAN

本练习包含以下主题：

- 监视 RMAN 作业
- 使用 EM 监视 RMAN

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.



诊断数据库

ORACLE®

版权所有 © 2011, 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：

- 检测和修复数据库损坏
- 处理块损坏
- 设置自动诊断资料档案库
- 运行健康检查

ORACLE®

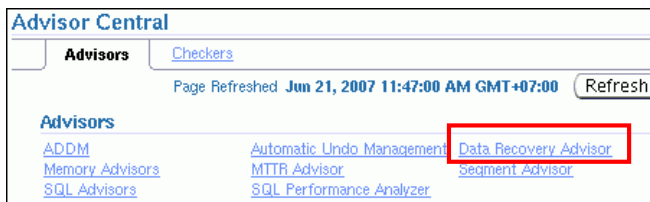
版权所有 © 2011, Oracle。保留所有权利。

数据恢复指导

> 数据恢复指导
块损坏
ADR
健康状况监视器

- 快速检测、分析和修复故障
- 将对用户的干扰降到最低
- 停机时和运行时的故障
- 用户界面：

- EM GUI 界面
(多个路径)
- RMAN 命令行



- 支持的数据库配置：
 - 单实例
 - 非 RAC
 - 支持故障转移到备用数据库，但不支持分析和修复备用数据库

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

数据恢复指导

在发生错误时，数据恢复指导可自动收集数据故障信息。此外，它还可以主动检查故障。在这种模式下，它有可能在数据库进程发现损坏并指出错误之前就检测和分析数据故障（请注意，修复始终由人控制。）

数据故障可能会很严重。例如，如果缺少最新的日志文件，则无法启动数据库。一些数据故障（如数据文件中的块损坏）不是灾难性故障，因为它们不会使数据库停机，也不会阻止您启动 Oracle 实例。数据恢复指导可以处理两种情况：一种情况是您无法启动数据库（因为缺少一些必需的数据库文件，或者这些数据库文件不一致或已损坏），另一种情况是运行时发现文件损坏。

用户界面

可从 Oracle Enterprise Manager (EM) Database Control 和 Grid Control 使用数据恢复指导。出现故障时，可使用多种方法访问数据恢复指导。以下示例都是从 “Database Instance（数据库实例）” 主页开始的：

- “Availability（可用性）” 选项卡页 > Perform Recover（执行恢复）> Advise and Recover（建议和恢复）
- 单击 “Support Workbench（支持工作台）” 的 “Problems（问题）” 页上的 “Active Incidents（活动意外事件）” 链接：“Checker Findings（检查器查找结果）” 选项卡页 > Launch Recovery Advisor（启动恢复指导）
- Database Instance Health（数据库实例健康状况）> 单击特定的链接，例如，“Incidents（意外事件）” 部分中的 ORA 1578 > “Support Workbench（支持工作台）” 的 “Problems Detail（问题详细资料）” 页 > Data Recovery Advisor（数据恢复指导）
- Database Instance Health（数据库实例健康状况）> “Related Links（相关链接）” 部分：Support Workbench（支持工作台）> “Checker Findings（检查器查找结果）” 选项卡页：Launch Recovery Advisor（启动恢复指导）
- Related Links（相关链接）：Advisor Central（指导中心）> “Advisors（指导）” 选项卡页：Data Recovery Advisor（数据恢复指导）
- Related Links（相关链接）：Advisor Central（指导中心）> “Checkers（检查器）” 选项卡页：Details（详细资料）> “Run Detail（运行详细资料）” 选项卡页：Launch Recovery Advisor（启动恢复指导）

也可以通过 RMAN 命令行使用它。例如：

```
rman target / nocatalog
rman> list failure all;
```

支持的数据库配置

在当前版本中，数据恢复指导支持单实例数据库，不支持 Oracle Real Application Clusters (RAC) 数据库。

数据恢复指导不能使用从备用数据库传送而来的块或文件修复主数据库中的故障。同时，您也无法使用数据恢复指导诊断并修复备用数据库上的故障。但是，数据恢复指导确实支持故障转移到备用数据库（作为修复方案，如上所述）。

数据恢复指导

通过理清混乱缩短停机时间：



ORACLE

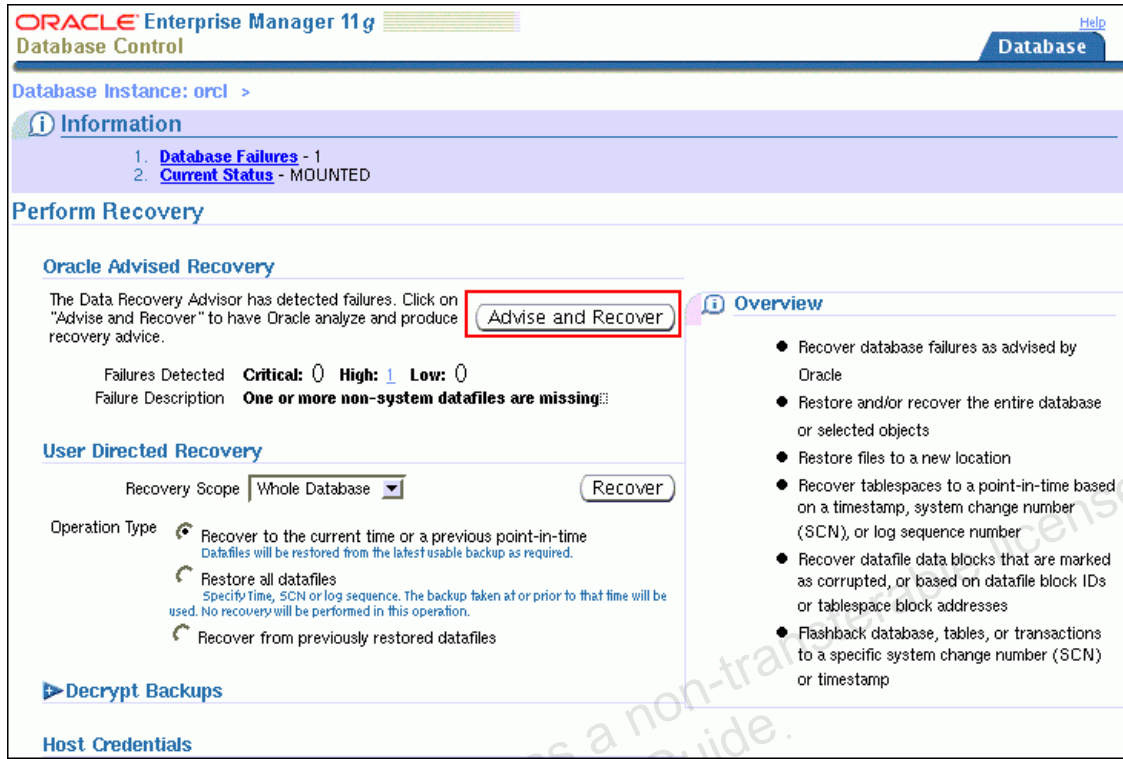
版权所有 © 2011, Oracle。保留所有权利。

数据恢复指导

Oracle Database 11g 中的自动诊断 workflow 按如下方式执行。使用数据恢复指导，您只需启动建议和修复。

1. 健康状况监视器会自动执行检查，并将故障及其故障现象作为“查找结果”记录到自动诊断资料档案库 (ADR) 中。
2. 数据恢复指导将查找结果与故障合并在一起，并按故障严重程度（严重或高）列出以前执行的评估结果。
3. 请求提供有关故障的修复建议时，数据恢复指导会将故障映射到自动和手动修复选项，检查基本可行性，然后提供修复建议。
4. 可以手动执行修复，也可请求数据恢复指导执行修复。
5. 除了健康状况监视器和数据恢复指导自动执行的主要“被动”检查之外，Oracle 还建议使用 VALIDATE 命令执行“主动”检查。

数据故障



版权所有 © 2011, Oracle。保留所有权利。

数据故障

可通过检查来检测数据故障，检查是评估数据库或其组件健康状况的诊断过程。每个检查可对一个或多个故障进行诊断，然后将其映射到修复。

检查可以是被动检查，也可以是主动检查。数据库中出现错误时，将自动执行“被动检查”。此外，您也可以启动“主动检查”，例如，执行 `VALIDATE DATABASE` 命令。

在 Oracle Enterprise Manager 中，如果发现数据库处于“停机”或“已装载”状态，请选择“Availability（可用性）> Perform Recovery（执行恢复）”，或单击“Perform Recovery（执行恢复）”按钮。

数据故障：示例

- 组件不可访问，例如：
 - 缺少操作系统级别的数据文件
 - 访问权限不正确
 - 表空间脱机等
- 物理损坏，如块校验和故障或块头字段值无效
- 逻辑损坏，如字典不一致、行片段损坏、索引条目损坏或事务处理损坏
- 不一致，如控制文件比数据文件和联机重做日志旧或新
- I/O 故障，如超过打开的文件数限制、通道不可访问、网络或 I/O 错误



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

数据故障：示例

数据恢复指导可以分析故障并建议问题修复选项，如幻灯片所述。

数据恢复指导 RMAN 命令行界面

RMAN 命令	操作
LIST FAILURE	列出先前执行的故障评估
ADVISE FAILURE	显示建议的修复选项
REPAIR FAILURE	修复并关闭故障（在 同一 RMAN 会话中的 ADVISE 命令后）
CHANGE FAILURE	更改或关闭一个或多个故障

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

数据恢复指导：RMAN 命令行界面

如果您怀疑或已知道数据库出现故障，则可使用 LIST FAILURE 命令获得这些故障的信息。可以列出所有故障或部分故障并以多种方式限制输出。故障由故障号唯一标识。请注意，这些故障号不是连续的，因此它们之间的间隔没有任何意义。

ADVISE FAILURE 命令将显示指定故障的建议修复选项。它打印输入故障概要并隐式关闭已修复的所有打开的故障。没有使用任何选项时，默认行为是对记录在 ADR 中优先级为 CRITICAL 和 HIGH 的所有故障提供建议。

REPAIR FAILURE 命令在**同一 RMAN 会话中的** ADVISE FAILURE 命令之后使用。默认情况下，该命令使用当前会话中上次执行 ADVISE FAILURE 命令时建议的一个修复选项。如果没有任何修复选项，REPAIR FAILURE 命令将启动隐式 ADVISE FAILURE 命令。修复完成后，该命令会关闭故障。

CHANGE FAILURE 命令将更改故障优先级或关闭一个或多个故障。仅能更改 HIGH 或 LOW 故障优先级。修复故障后，将隐式关闭打开的故障。不过，您也可以显式关闭故障。

列出数据故障

RMAN LIST FAILURE 命令列出先前执行的故障评估。

- 包括新诊断的故障
- 删除关闭的故障（默认情况下）



语法：

```
LIST FAILURE
[ ALL | CRITICAL | HIGH | LOW | CLOSED |
  failnum[,failnum,...] ]
[ EXCLUDE FAILURE failnum[,failnum,...] ]
[ DETAIL ]
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

列出数据故障

RMAN LIST FAILURE 命令列出故障。如果目标实例使用恢复目录，它可以处于 **STARTED** 模式下，否则必须处于 **MOUNTED** 模式下。**LIST FAILURE** 命令不启动检查来诊断新故障；它将列出先前执行的评估的结果。重复执行 **LIST FAILURE** 命令可重新验证所有现有的故障。如果数据库诊断出新的故障（在命令执行之间），则会显示这些新故障。如果用户手动修复故障或临时故障消失，则数据恢复指导会将这些故障从 **LIST FAILURE** 输出中删除。以下是语法说明：

- **failnum**：要为其显示修复选项的故障数。
- **ALL**：列出所有优先级的故障。
- **CRITICAL**：列出优先级为 **CRITICAL** 且状态为 **OPEN** 的故障。这些故障（如控制文件缺失）使整个数据库不可用，因此需要立即进行关注。
- **HIGH**：列出优先级为 **HIGH** 且状态为 **OPEN** 的故障。这些故障（如归档重做日志缺失）使数据库部分不可用或不可恢复，因此应尽快修复。
- **LOW**：列出优先级为 **LOW** 且状态为 **OPEN** 的故障。低优先级的故障可以等到修复了更重要的故障后再进行修复。

列出数据故障（续）

- CLOSED: 仅列出关闭的故障。
- EXCLUDE FAILURE: 从列表中排除指定的故障号。
- DETAIL: 列出故障并展开合并的故障。例如，如果一个文件中有多个块损坏，则 DETAIL 选项将列出每个块损坏。

有关命令语法的详细资料，请参阅《Oracle Database Backup and Recovery Reference》。

修复建议

RMAN ADVISE FAILURE 命令：

- 显示输入故障列表概要
- 包括警告（如果 ADR 中出现新故障）
- 显示手动核对清单
- 列出一个建议的修复选项
- 生成修复脚本（用于自动或手动修复）

```
. . .
Repair script:
  /u01/app/oracle/diag/rdbms/orcl/orcl/hm/reco_2979
  128860.hm
RMAN>
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

修复建议

RMAN ADVISE FAILURE 命令显示针对指定故障建议的修复选项。ADVISE FAILURE 命令打印输入故障概要。该命令会隐式关闭已修复的所有打开的故障。

默认行为（没有使用任何选项时）是对自动诊断资料档案库 (ADR) 中记录的 CRITICAL 和 HIGH 优先级的所有故障提供建议。如果自上次执行 LIST FAILURE 命令后 ADR 中记录了新故障，则在对所有 CRITICAL 和 HIGH 故障提供建议前，该命令将包含一个 WARNING。

可执行两个常规修复选项：无数据丢失修复和数据丢失修复。

数据恢复指导在生成自动修复选项时会生成一个脚本，用于显示 RMAN 计划如何修复故障。如果不希望数据恢复指导自动修复故障，可从该脚本开始执行手动修复。该脚本的操作系统 (OS) 位置将显示在命令输出的末尾。您可以检查此脚本，并对其进行自定义（如果需要），还可以手动执行该脚本（例如，您的审计线索要求建议执行手动操作时）。

语法

```
ADVISE FAILURE
[ ALL | CRITICAL | HIGH | LOW | failnum[,failnum,...] ]
[ EXCLUDE FAILURE failnum [,failnum,...] ]
```

执行修复

RMAN REPAIR FAILURE 命令：

- 在 ADVISE FAILURE 命令之后执行
- 修复指定故障
- 关闭修复的故障

语法：

```
REPAIR FAILURE  
  
[USING ADVISE OPTION integer]  
  
[ { {NOPROMPT | PREVIEW}}...]
```



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

执行修复

此命令应在同一 RMAN 会话中在 ADVISE FAILURE 命令之后使用。默认情况下（没有指定任何选项时），该命令使用当前会话中上次执行 ADVISE FAILURE 时建议的一个修复选项。如果没有任何修复选项，REPAIR FAILURE 命令将启动隐式 ADVISE FAILURE 命令。

使用 USING ADVISE OPTION *integer*，可按选项编号指定所需的修复选项（从 ADVISE FAILURE 命令中）；这不是故障号。

默认情况下，您需要确认是否执行该命令，因为可能需要花费许多时间完成大量更改。在执行修复期间，该命令的输出指明正在执行的修复阶段。

修复完成后，该命令会关闭故障。

无法运行多个并发修复会话。但是，可以运行并发 REPAIR ... PREVIEW 会话。

- PREVIEW：不执行修复，而是显示先前生成的 RMAN 脚本以及所有修复操作和注释。
- NOPROMPT：不要求确认。

分类（和关闭）故障

RMAN CHANGE FAILURE 命令：

- 更改故障优先级（CRITICAL 除外）
- 关闭一个或多个故障

示例：

```
RMAN> change failure 5 priority low;
List of Database Failures
=====
Failure ID Priority Status      Time Detected Summary
-----
5          HIGH      OPEN        20-DEC-06    one or more
             datafiles are missing
Do you really want to change the above failures (enter YES or
NO)? yes
changed 1 failures to LOW priority
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

分类（和关闭）故障

CHANGE FAILURE 命令用于更改故障优先级或关闭一个或多个故障。

语法

```
CHANGE FAILURE
{ ALL | CRITICAL | HIGH | LOW | failnum[,failnum,...] }
[ EXCLUDE FAILURE failnum[,failnum,...] ]
{ PRIORITY {CRITICAL | HIGH | LOW} |
CLOSE } — 将故障的状态更改为关闭
[ NOPROMPT ] — 不要求用户确认
```

只能将故障优先级从 HIGH 更改为 LOW，以及从 LOW 更改为 HIGH。更改 CRITICAL 优先级是错误做法。（将故障的优先级从 HIGH 更改为 LOW 的一个原因是避免该故障显示在 LIST FAILURE 命令的默认输出列表中。例如，如果某个损坏的块具有 HIGH 优先级，且该块位于很少使用的表空间中，您可能希望将其临时更改为 LOW。）

修复故障后，将隐式关闭打开的故障。不过，您也可以显式关闭故障。这需要重新评估其它所有打开的故障，因为其中的某些故障会因该故障的关闭而变得不相关。

默认情况下，该命令要求用户确认请求的更改。

数据恢复指导视图

查询 V\$ 视图：

- V\$IR_FAILURE：所有故障的列表，其中包括已关闭的故障（LIST FAILURE 命令的结果）
- V\$IR_MANUAL_CHECKLIST：手动建议的列表（ADVISE FAILURE 命令的结果）
- V\$IR_REPAIR：修复的列表（ADVISE FAILURE 命令的结果）
- V\$IR_FAILURE_SET：故障和建议标识符的交叉引用



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

数据恢复指导视图

有关数据恢复指导使用的动态数据字典视图的详细资料，请参阅《Oracle Database Reference》。

最佳做法：主动检查

调用对数据库及其组件的主动健康检查：

- 健康状况监视器或 `RMAN VALIDATE DATABASE` 命令
- 检查逻辑和物理损坏
- 在 ADR 中记录查找结果



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

最佳做法：主动检查

对于非常重要的数据库，您可能需要执行其它主动检查（可以在每天的低峰时段执行）。可通过健康状况监视器或使用 `RMAN VALIDATE` 命令安排定期的健康检查。通常，如果被动检查在数据库组件中检测到故障，则可能需要对受影响的组件执行更全面的检查。

`RMAN VALIDATE DATABASE` 命令用于调用对数据库及其组件的健康检查。它扩展了现有 `VALIDATE BACKUPSET` 命令。在验证期间检测到的所有问题都会显示出来，这些问题进而会启动故障评估。如果检测到故障，则该故障会作为查找结果记录到 ADR 中。可以使用 `LIST FAILURE` 命令查看资料档案库中记录的所有故障。

`VALIDATE` 命令支持对单个备份集和数据块进行验证。在物理损坏中，数据库根本无法识别块。在逻辑损坏中，块的内容在逻辑上不一致。默认情况下，`VALIDATE` 命令只检查物理损坏。也可以指定 `CHECK LOGICAL` 来检查逻辑损坏。

块损坏可分为块间损坏和块内损坏。在块内损坏中，块本身发生损坏，可能是物理损坏也可能是逻辑损坏。在块间损坏中，块与块之间发生的损坏只能是逻辑损坏。`VALIDATE` 命令只检查块内损坏。

什么是块损坏

- 只要对块执行读或写操作，就会执行下列一致性检查。
 - 块版本
 - 高速缓存中的 DBA（数据块地址）值与块缓冲区中的 DBA 值比较的结果
 - 块校验和（如果启用）
- 损坏的块被标识为以下类别：
 - 介质损坏
 - 逻辑（或软件）损坏

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

什么是块损坏

所谓损坏的数据块，是指块没有采用可识别的 Oracle 格式，或者其内容在内部不一致。通常情况下，损坏是由硬件故障或操作系统问题引起的。Oracle DB 将损坏的块标识为“逻辑损坏”或“介质损坏”。如果是逻辑损坏，则是 Oracle 内部错误。Oracle DB 检测到不一致之后，就将逻辑损坏的块标记为损坏。如果是介质损坏，则是块格式不正确；从磁盘读取的块没有包含有意义的信息。

正如您刚才学习的，可以使用数据恢复指导修复一些数据故障和损坏。现在您将学习手动诊断和修复损坏的方法。

通过恢复块，或者删除包含损坏块的数据库对象（或同时采用这两种方式），可以修复介质损坏的块。如果介质损坏是由硬件故障引起的，则只有修复了硬件故障后，才能彻底解决问题。

块损坏故障现象：ORA-01578

ORA-01578 错误: "ORACLE data block corrupted (file # %s, block # %s)":

- 发现损坏的数据块时生成此信息
- 始终返回相对文件号和相对块号
- 返回到发出查询的会话（该查询在发现损坏时执行）
- 显示在 alert.log 文件中

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

块损坏故障现象：ORA-01578

一般情况下，ORA-01578 错误是由硬件问题引起的。如果 ORA-01578 错误始终返回相同的参数，则最可能的原因是块介质损坏。

如果返回的参数每次都有变化，则可能存在硬件问题。应检查内存和页面空间，并检查 I/O 子系统，以便查找有问题的控制器。

注：ORA-01578 返回相对文件号，但伴随出现的 ORA-01110 错误会显示绝对文件号。

如何处理损坏

- 检查预警日志和操作系统日志文件。
- 使用可用的诊断工具，找出损坏的类型。
- 多次运行检查功能，确定错误是否持续存在。
- 根据需要，从损坏的对象中恢复数据。
- 解决硬件问题：
 - 内存条
 - 磁盘控制器
 - 磁盘
- 根据需要，从损坏的对象中恢复或还原数据。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

如何处理损坏

始终尝试确定错误是否持续出现。多次运行 ANALYZE 命令；如果可能，可执行关闭再启动操作，然后再次尝试早先发生故障的操作。查找是否有其它损坏。如果发现一个损坏的块，则可能还存在其它损坏的块。

硬件故障必须立即解决。遇到硬件问题时，应与供应商取得联系，在检查并修复了计算机后再继续工作。此时应运行一次全面的硬件诊断会话。

硬件故障的类型可能会有很多种：

- I/O 硬件或固件故障
- 操作系统 I/O 或高速缓存问题
- 内存或分页问题
- 磁盘修复实用程序

设置检测损坏的参数

Name	Help	Revisions	Value	Comments	Type	Basic	Modified	Dynamic	Category
db_block_checking			FALSE	防止内存和数据损坏				✓	Diagnostics and Statistics
db_block_checksum			FALSE		String			✓	Diagnostics and Statistics
db_block_size			LOW		Integer	✓	✓		Memory
db_block_size			MEDIUM		Integer	✓	✓		Memory
db_block_size			TRUE		String			✓	Memory
db_block_size			FULL		String			✓	Memory
db_block_checksum			TYPICAL	检测 I/O 存储、磁盘损坏				✓	Diagnostics and Statistics
db_block_checksum			OFF		String			✓	Diagnostics and Statistics
db_block_size			FALSE		Integer	✓	✓		Memory
db_block_size			TYPICAL		String			✓	Memory
db_block_size			TRUE		String			✓	Memory
db_block_size			FULL		Big			✓	Memory
db_lost_write_protect			TYPICAL	检测物理备用数据库上的非持久写入					
db_name			NONE		String	✓	✓		Database Identification
db_name			TYPICAL		String				Database Identification
db_name			FULL		String				Database Identification
db_ultra_safe			OFF	为损坏检测指定默认值					Miscellaneous

EM > Server (服务器) > Initialization Parameters (初始化参数)

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

设置检测损坏的参数

可以使用 DB_ULTRA_SAFE 参数来简化管理。它会影响下列参数的默认值：

- DB_BLOCK_CHECKING，用于启动对数据库块的检查。此检查通常可防止内存和数据损坏。（默认值：FALSE，建议值：FULL）
- DB_BLOCK_CHECKSUM，将每个数据块写入到磁盘时，在其高速缓存标头中启动校验和的计算和存储。校验和有助于检测由基础磁盘、存储系统或 I/O 系统引起的损坏。（默认值：TYPICAL，建议值：TYPICAL）
- DB_LOST_WRITE_PROTECT，用于启动对“丢失的写入”的检查。如果 I/O 子系统指示块写入已完成，但该块尚未完全写入到持久存储中，则物理备用数据库上会发生数据块写入丢失。当然，写入操作在主数据库中已完成。（默认值：TYPICAL，建议值：TYPICAL）

如果显式设置了其中任何一个参数，则您设置的值将保持有效。DB_ULTRA_SAFE 参数（Oracle Database 11g 中的新增参数）仅更改这些参数的默认值。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

设置检测损坏的参数

DB_ULTRA_SAFE	OFF	DATA_ONLY	DATA_AND_INDEX
DB_BLOCK_CHECKING	OFF 或 FALSE	MEDIUM	FULL 或 TRUE
DB_BLOCK_CHECKSUM	TYPICAL	FULL	FULL
DB_LOST_WRITE_PROTECT	TYPICAL	TYPICAL	TYPICAL

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

设置检测损坏的参数（续）

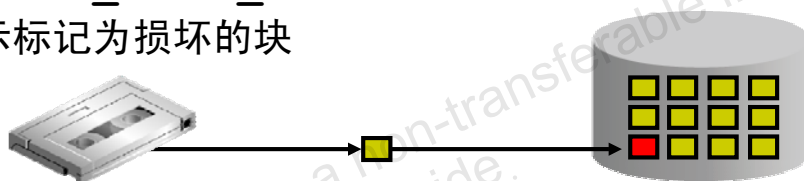
根据系统对块损坏的容忍度，可以加强对块损坏的检查。启用 DB_ULTRA_SAFE 参数（默认值：OFF）会导致系统开销增加，因为这些检查更加密集。开销量与每秒更改的块数相关，所以无法进行简单量化。对于“高更新的”应用程序，CPU 使用率会显著增加，大约在 10% 到 20% 之间，不过可能更高。通过分配额外的 CPU 可减轻此开销。

- 当 DB_ULTRA_SAFE 参数设置为 DATA_ONLY 时，DB_BLOCK_CHECKING 参数设置为 MEDIUM。这会检查块中的数据在逻辑上是否一致。内存中的块内容发生更改后（例如，在执行 UPDATE 或 INSERT 命令、执行磁盘上读取或在 Oracle RAC 中实例间传送块后），将执行基本的块标头检查。此级别的检查包括对非索引表块进行的语义块检查。
- 当 DB_ULTRA_SAFE 参数设置为 DATA_AND_INDEX 时，DB_BLOCK_CHECKING 参数设置为 FULL。除了上面的检查外，还会执行语义检查以查找索引块（即遇到损坏时，可实际删除和重新构造从属对象的块）。
- 当 DB_ULTRA_SAFE 参数设置为 DATA_ONLY 或 DATA_AND_INDEX 时，DB_BLOCK_CHECKSUM 参数设置为 FULL，DB_LOST_WRITE_PROTECT 参数设置为 TYPICAL。

块介质恢复

块介质恢复：

- 降低平均恢复时间 (MTTR)
- 提高介质恢复期间的可用性
 - 恢复期间数据文件保持联机状态
 - 只有正在恢复的块是不可访问的
- 是使用 **RMAN RECOVER...BLOCK** 命令调用的
 - 使用闪回日志以及完全备份或 0 级备份还原块
 - 介质恢复是使用重做日志执行的
- **V\$DATABASE_BLOCK_CORRUPTION** 视图显示标记为损坏的块



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

块介质恢复

大多数情况下，第一次遇到损坏时，数据库会将块标记为介质损坏，然后将其写到磁盘上。在该块得到恢复之前，不能对其执行任何后续读取操作。只能对标记为损坏或者未通过损坏检查的块执行块恢复。块介质恢复是使用 **RMAN RECOVER...BLOCK** 命令执行的。默认情况下，RMAN 会搜索闪回日志以查找好的块副本，然后在完全备份或 0 级增量备份中搜索块。如果 RMAN 找到了好的副本，则会还原这些副本，并对块执行介质恢复。块介质恢复只能使用重做日志进行介质恢复，不能使用增量备份。

V\$DATABASE_BLOCK_CORRUPTION 视图显示由数据库组件（如 RMAN 命令、ANALYZE、dbv、SQL 查询等）标记为损坏的块。对于以下类型的损坏此视图会增加相应的行：

- **物理/介质损坏：**数据库无法识别块：校验和无效、块内容全部为零或者块标头不完整。物理损坏检查是默认启用的。
- **逻辑损坏：**块的校验和有效，块头和块尾也匹配，但是内容不一致。块介质恢复不能修复逻辑块损坏。默认情况下，逻辑损坏检查处于禁用状态。通过指定 **BACKUP**、**RESTORE**、**RECOVER** 和 **VALIDATE** 命令的 **CHECK LOGICAL** 选项，可以启用逻辑损坏检查。

块介质恢复的先决条件

- 目标数据库必须处于 ARCHIVELOG 模式。
- 包含损坏块的数据文件的备份必须是完全备份或 0 级备份。
 - 要使用代理副本，必须先将它们还原到非默认位置。
- RMAN 只能使用归档重做日志进行恢复。
- 可从闪回日志（如果可用）还原损坏的数据块。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

块介质恢复的先决条件

下列先决条件适用于 RECOVER ... BLOCK 命令：

- 目标数据库必须以 ARCHIVELOG 模式运行，并且必须是打开的，或是使用当前控制文件装载的。
- 包含损坏块的数据文件备份必须是完全备份或 0 级备份，不能是代理副本。如果只存在代理副本备份，则可将它们还原到磁盘上的非默认位置；在这种情况下，RMAN 会认为它们是数据文件副本，在块介质恢复过程中会在其中搜索块。
- RMAN 只能使用归档重做日志进行恢复。RMAN 不能使用 1 级增量备份。块介质恢复不能恢复丢失或无法访问的归档重做日志，但有时可以恢复丢失的重做记录。
- 必须在目标数据库上启用闪回数据库，这样 RMAN 才能在闪回日志中搜索损坏块的好副本。如果启用了闪回日志记录，而且该日志记录包含损坏块的较旧但未损坏的版本，则 RMAN 可以使用这些块，因而可能会提高恢复的速度。

RECOVER...BLOCK 命令

RMAN RECOVER...BLOCK 命令:

- 确定包含要进行恢复的块的备份
- 读取备份并将请求的块累积到内存缓冲区
- 必要时, 通过从备份中读取归档日志来管理块介质恢复会话

```
RECOVER DATAFILE 6 BLOCK 3; Recover a single block

RECOVER                                Recover multiple blocks
DATAFILE 2 BLOCK 43                    in multiple data files
DATAFILE 2 BLOCK 79
DATAFILE 6 BLOCK 183;

RECOVER CORRUPTION LIST; Recover all blocks logged in
                        V$DATABASE_BLOCK_CORRUPTION
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

恢复单个块

在进行块恢复之前, 必须确定损坏的块。一般情况下, 会在以下位置中报告块损坏:

- LIST FAILURE、VALIDATE 或 BACKUP ... VALIDATE 命令的结果
- V\$DATABASE_BLOCK_CORRUPTION 视图
- 标准输出中的错误消息
- 预警日志文件和用户跟踪文件 (在 V\$DIAG_INFO 视图中标识)
- SQL ANALYZE TABLE 和 ANALYZE INDEX 命令的结果
- DBVERIFY 实用程序的结果

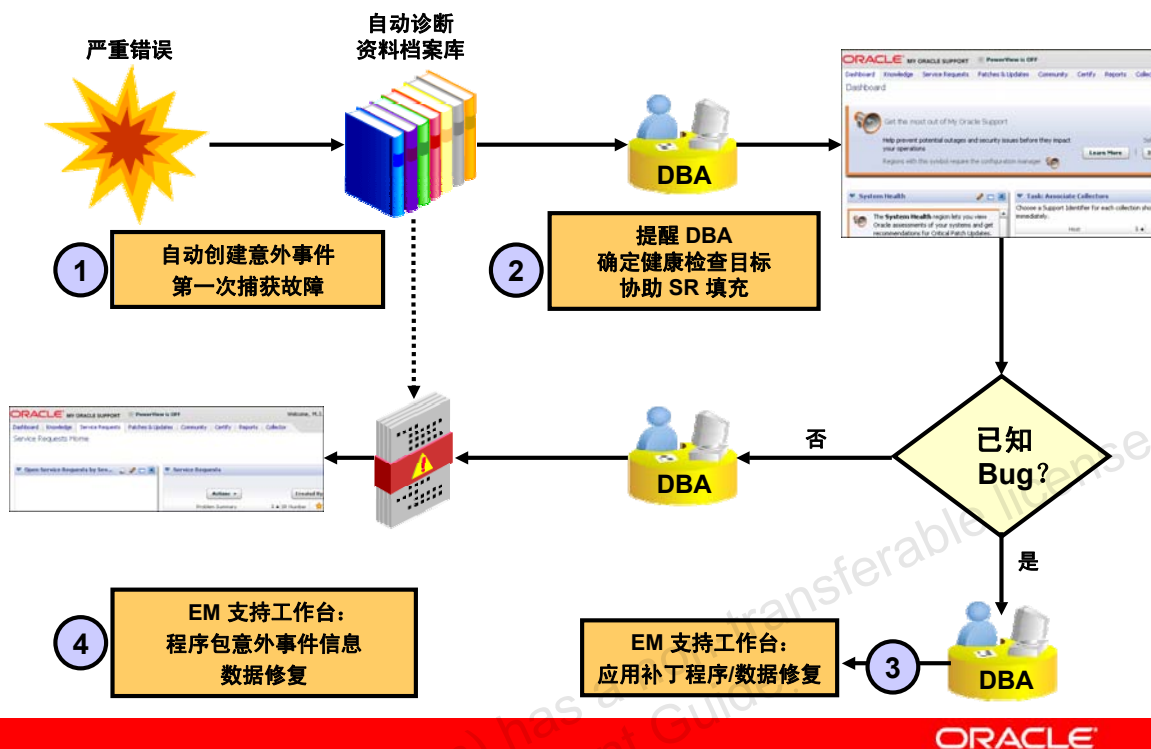
例如, 可能会在用户跟踪文件中发现以下消息:

```
ORA-01578: ORACLE data block corrupted (file # 7, block # 3)
ORA-01110: data file 7: '/oracle/oradata/orcl/tools01.dbf'
ORA-01578: ORACLE data block corrupted (file # 2, block # 235)
ORA-01110: data file 2: '/oracle/oradata/orcl/undotbs01.dbf'
```

确定了块以后, 在 RMAN 提示符下运行 RECOVER ... BLOCK 命令, 指定损坏块的文件号和块号。

```
RECOVER
DATAFILE 7 BLOCK 3
DATAFILE 2 BLOCK 235;
```

自动诊断 workflow



版权所有 © 2011, Oracle. 保留所有权利。

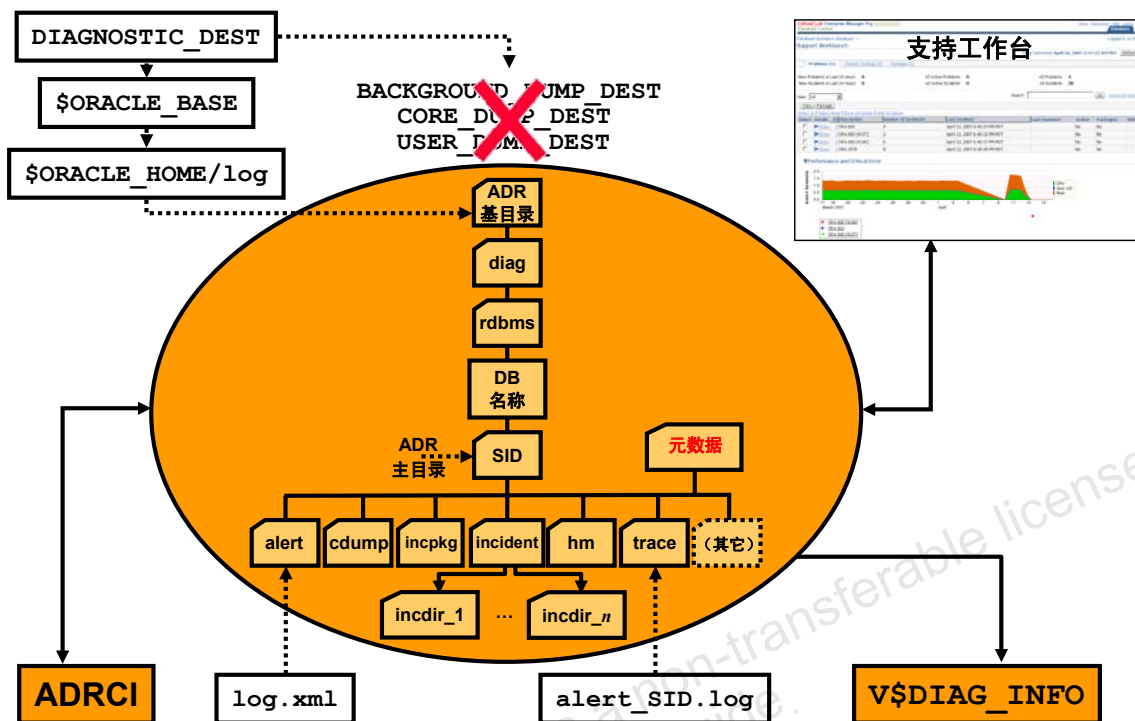
自动诊断 workflow

通过一个始终处于打开状态的内存中跟踪工具，数据库组件可以在第一次出现严重错误故障时捕获诊断数据。系统将自动维护一个称为“自动诊断资料档案库”的特殊资料档案库，以保存有关严重错误事件的诊断信息。此信息可用于创建要发送到 Oracle 支持服务以进行调查的意外事件程序包。

下面是一个典型的诊断会话 workflow:

1. 意外事件导致 Oracle Enterprise Manager (EM) 中出现预警。
2. DBA 可通过 EM 的“Alert (预警)”页查看此预警。
3. DBA 可以细化到意外事件和问题的详细资料。
4. DBA 或 Oracle 支持服务可以决定或要求将信息打包并通过 My Oracle Support 发送到 Oracle 支持服务。DBA 可以将文件添加到自动打包的数据中。

自动诊断资料档案库



版权所有 © 2011, Oracle。保留所有权利。

自动诊断资料档案库 (ADR)

ADR 是一个基于文件的资料档案库，用于存放数据库诊断数据（如跟踪、意外事件转储和程序包、预警日志、健康状况监视器报表、核心转储等）。它对存储在任何数据库外的多个实例和多种产品使用一个统一的目录结构。因此，在数据库关闭时可用来诊断问题。

从 Oracle Database 11g R1 开始，数据库、自动存储管理 (ASM)、集群就绪服务 (CRS) 和其它 Oracle 产品或组件将所有诊断数据都存储在 ADR 中。每种产品的每个实例都将诊断数据存储在各自的 ADR 主目录下。例如，在具有共享存储和 ASM 的 Real Application Clusters 环境中，每个数据库实例和每个 ASM 实例在 ADR 中都有一个主目录。利用 ADR 的统一目录结构、用于各种产品和实例的统一诊断数据格式以及一组统一的工具，客户和 Oracle 技术支持部门可以在多个实例间关联并分析诊断数据。

ADR 根目录又称为 ADR 基目录，其位置由 `DIAGNOSTIC_DEST` 初始化参数设置。如果此参数被忽略或留为空值，则数据库在启动时将对 `DIAGNOSTIC_DEST` 进行如下设置：如果设置了环境变量 `ORACLE_BASE`，则将 `DIAGNOSTIC_DEST` 设置为 `$ORACLE_BASE`。如果未设置环境变量 `ORACLE_BASE`，则将 `DIAGNOSTIC_DEST` 设置为 `$ORACLE_HOME/log`。

ADR 命令行工具 (ADRCI)

- ADRCI 可以通过操作系统提示符与 ADR 进行交互。
- 使用 ADRCI 可以查看自动诊断资料档案库中的诊断数据。

```
$ adrci
ADRCI: Release 11.1.0.5.0 - On Sat Jul 7 08:01:40 2007
Copyright (c) 1982, 2007, Oracle. All rights reserved.

ADR base = "/u01/app/oracle"

ADRCI> show incident
ADR Home = /u01/app/oracle/product/11.1.0/db_1/log/diag/rdbms/orcl/orcl:
*****
INCIDENT_ID PROBLEM_KEY                                CREATE_TIME
-----
1681          ORA-600_dbgris01:1,_addr=0xa9876541  17-JAN-07 09.17.44.843125...
1682          ORA-600_dbgris01:12,_addr=0xa9876542 18-JAN-07 09.18.59.434775...
2 incident info records fetched
```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

ADR 命令行工具 (ADRCI)

ADRCI 是一种命令行工具，属于数据库故障诊断基础结构的一部分。使用 ADRCI 可以：

- 查看自动诊断资料档案库 (ADR) 中的诊断数据
- 将意外事件和问题信息打包成 zip 文件传输到 Oracle 技术支持部门

ADRCI 可在交互模式下使用，也可以在脚本中使用。此外，ADRCI 还可以执行 ADRCI 命令脚本，使用的方式与 SQL*Plus 执行 SQL 脚本和 PL/SQL 命令的方式相同。由于未打算保护 ADR 中数据的安全，所以没有必要登录到 ADRCI。对 ADR 目录的操作系统权限是针对 ADR 数据的唯一保护机制。

对诊断数据进行打包和其它管理的最简单方法是使用 Oracle Enterprise Manager 的支持工作台（帮助解析数据库错误以及 ASM 错误）。

ADRCI 提供了一个可代替支持工作台大部分功能的命令行，并添加了一些功能，如列出和查询跟踪文件。本幻灯片中的示例显示了一个 ADRCI 会话，该会话列出了存储在 ADR 中的所有打开的意外事件。

注：有关 ADRCI 和支持工作台的详细信息，请参阅《Oracle Database Utilities》指南。

V\$DIAG_INFO 视图

```
SQL> SELECT * FROM V$DIAG_INFO;
```

NAME	VALUE
Diag Enabled	TRUE
ADR Base	/u01/app/oracle
ADR Home	/u01/app/oracle/diag/rdbms/orcl/orcl
Diag Trace	/u01/app/oracle/diag/rdbms/orcl/orcl/trace
Diag Alert	/u01/app/oracle/diag/rdbms/orcl/orcl/alert
Diag Incident	/u01/app/oracle/diag/rdbms/orcl/orcl/incident
Diag Cdump	/u01/app/oracle/diag/rdbms/orcl/orcl/cdump
Health Monitor	/u01/app/oracle/diag/rdbms/orcl/orcl/hm
Default Trace File	/u01/app/oracle/diag/.../trace/orcl_ora_11424.trc
Active Problem Count	3
Active Incident Count	8

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

V\$DIAG_INFO 视图

V\$DIAG_INFO 视图列出了所有重要 ADR 位置：

- ADR Base: ADR 基目录的路径。
- ADR Home: 当前数据库实例的 ADR 主目录的路径。
- Diag Trace: 文本预警日志和后台/前台进程跟踪文件的位置。
- Diag Alert: XML 版本的预警日志的位置。
- Diag Incident: 意外事件日志的写入位置。
- Diag Cdump: 诊断核心文件写入到此目录中。
- Health Monitor: 运行健康状况监视器时产生的日志的位置。
- Default Trace File: 会话的跟踪文件的路径。SQL 跟踪文件写入到此位置。

诊断跟踪的位置

诊断数据	以前的位置	ADR 位置
前台进程跟踪	USER_DUMP_DEST	ADR_HOME/trace
后台进程跟踪	BACKGROUND_DUMP_DEST	ADR_HOME/trace
预警日志数据	BACKGROUND_DUMP_DEST	ADR_HOME/alert ADR_HOME/trace
核心转储	CORE_DUMP_DEST	ADR_HOME/cdump
意外事件转储	USER BACKGROUND_DUMP_DEST	ADR_HOME/incident/incdir_n

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

诊断跟踪的位置

幻灯片中的表比较了 Oracle Database 10g 与 Oracle Database 11g 中都存在的各类跟踪数据和转储。

在 Oracle Database 11g 中，前台和后台跟踪文件的存储位置没有什么区别。这两种类型的文件都会放入 `ADR_HOME/trace` 目录中。

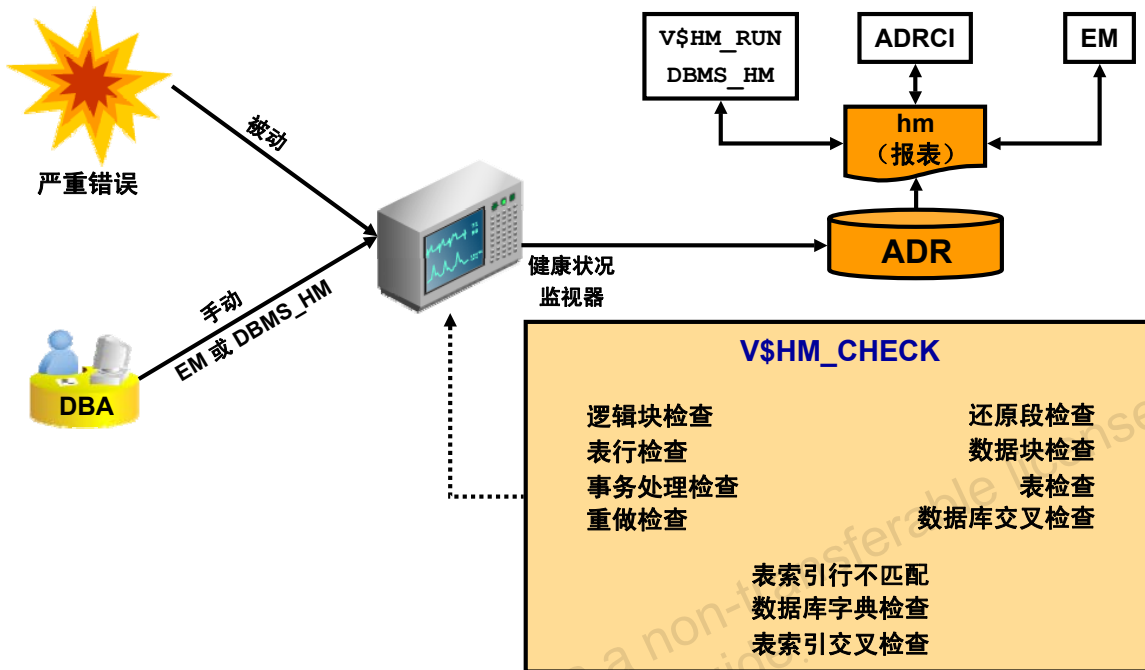
所有非意外事件跟踪都存储在 `trace` 子目录中。以前的版本会将严重错误信息转储到相应的进程跟踪文件而不是意外事件转储，这就是新旧版本之间的主要区别。从 Oracle Database 11g 开始，意外事件转储存放于独立于普通进程跟踪文件的文件中。

跟踪和转储之间的主要区别在于，跟踪是较为连续的输出（如打开 SQL 跟踪时），而转储是为了响应事件（如意外事件）而进行的一次性输出。另外，核心是特定于端口的二进制内存转储。

注：在幻灯片中，`ADR_HOME` 表示路径 `/u01/app/oracle/diag/rdbms/orcl/orcl`（假定实例名称为 `orcl`）。但是，不存在名为 `ADR_HOME` 的正式环境变量。

健康装款监视器：概览

数据恢复指导
块损坏
ADR
> 健康状况监视器



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

健康装款监视器

Oracle DB 包含一个称为健康状况监视器的框架，用于对数据库的各个组件运行诊断检查。健康状况监视器可以检查数据库的各个组件（包括文件、内存、事务处理完整性、元数据和进程使用情况）。这些检查将生成查找结果报表以及解决问题的建议。故障诊断基础结构可以在严重错误发生后自动运行健康状况监视器检查；另外，DBA 可以使用 DBMS_HM PL/SQL 程序包或 Oracle Enterprise Manager 界面手动运行健康状况监视器健康检查。

有关健康状况监视器可运行的所有可能检查的完整说明，请参考 V\$HM_CHECK。这些健康检查分为两个类别：

- **DB 联机：**可以在数据库处于打开状态（即处于 OPEN 模式）时运行这些检查。
- **DB 脱机：**除在数据库处于打开状态时可运行这些检查以外，还可在实例可供使用但数据库本身处于关闭状态（即处于 NOMOUNT 模式）时运行。

运行检查器后，将生成一份报表。此报表包含有关检查器调查结果的信息，包括优先级（低、高或严重），调查及其结果的说明，以及有关执行的基本统计信息。健康状况监视器可生成 XML 格式的报表并将其存储在 ADR 中。可使用 V\$HM_RUN、DBMS_HM、ADRCI 或 Oracle Enterprise Manager 查看这些报表。

手动运行健康检查：PL/SQL 示例

```
SQL> exec dbms_hm.run_check('Database Dictionary Check',
                           'mycheck',0,'TABLE_NAME=tab$');

SQL> set long 100000
SQL> select dbms_hm.get_run_report('mycheck') from dual;

DBMS_HM.GET_RUN_REPORT('mycheck')
-----
<?xml version="1.0" encoding="US-ASCII"?>
<HM-REPORT REPORT_ID="mycheck"><TITLE>HM Report: mycheck</TITLE>
  <RUN_INFO>
    <CHECK_NAME>Database Dictionary Check</CHECK_NAME>
    <RUN_ID>21</RUN_ID><RUN_NAME>mycheck</RUN_NAME>
    <RUN_MODE>MANUAL</RUN_MODE><RUN_STATUS>COMPLETED</RUN_STATUS> ...
  </RUN_INFO>
  <RUN_PARAMETERS><RUN_PARAMETER>TABLE_NAME=tab$</RUN_PARAMETER> ... </RUN_PARAMETERS>
  <RUN-FINDINGS><FINDING>
    <FINDING_NAME>Dictionary Inconsistency</FINDING_NAME><FINDING_ID>22</FINDING_ID>
    <FINDING_TYPE>FAILURE</FINDING_TYPE><FINDING_STATUS>OPEN</FINDING_STATUS>
    <FINDING_PRIORITY>CRITICAL</FINDING_PRIORITY> ...
    <FINDING_CREATION_TIME>...</FINDING_CREATION_TIME>
    <FINDING_MESSAGE>...invalid column number 7 on Object tab$ Failed</FINDING_MESSAGE>
    <FINDING_MESSAGE>Damaged ... Object SH.JFVTEST is referenced </FINDING_MESSAGE> ...
```

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

手动运行健康检查：PL/SQL 示例

可以使用 DBMS_HM.RUN_CHECK 过程运行健康检查。要调用 RUN_CHECK，请提供检查的名称（可在 V\$HM_CHECK 中找到）、运行名称（就是稍后用于检索报表的标签）和控制其执行的一组相应输入参数。可使用 V\$HM_CHECK_PARAM 查看这些参数。

在幻灯片的示例中，需要对 TAB\$ 表运行数据库字典检查（假定有一个重要的核心字典对象）。您将此运行命名为 MYCHECK，而且不希望为此检查设置任何超时。

执行时，您将执行 DBMS_HM.GET_RUN_REPORT 函数获取从 V\$HM_RUN、V\$HM_FINDING 和 V\$HM_RECOMMENDATION 中提取的报表。输出明确地显示了在 TAB\$ 中发现了严重错误。此表包含一个列数无效的表的条目。此外，报表还提供 TAB\$ 中损坏的表的名称。

调用 GET_RUN_REPORT 函数时，会在 ADR 的 HM 目录中生成 XML 报表文件。在示例中，该文件称为 HMREPORT_mycheck.hm。

注：有关 DBMS_HM 的详细信息，请参阅《Oracle Database PL/SQL Packages and Types Reference》。

使用 ADRCI 实用程序查看 HM 报表

```

adrci>>show hm_run
...
-----
RUN_ID                11081
RUN_NAME              HM_RUN_11081
CHECK_NAME            Database Cross Check
NAME_ID              2
MODE                 2
START_TIME            2007-04-13 03:20:31.161396 -07:00
RESUME_TIME
END_TIME              2007-04-13 03:20:37.903984 -07:00
MODIFIED_TIME        2007-04-17 01:16:37.106344 -07:00
TIMEOUT              0
FLAGS                0
STATUS               5
SRC_INCIDENT_ID      0
NUM_INCIDENTS        0
ERR_NUMBER           0
REPORT_FILE
...
adrci>>create report hm_run HM_RUN_11081
Adrci>>show report hm_run HM_RUN_11081
...

```

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 ADRCI 实用程序查看 HM 报表

可以使用 ADRCI 实用程序创建和查看健康状况监视器检查器报表。要执行该操作，请确保操作系统环境变量（如 ORACLE_HOME）设置正确，然后在操作系统命令提示符下输入以下命令：adrci。

该实用程序会启动，并显示幻灯片中所显示的提示符。可以选择更改当前的 ADR 主目录。使用 SHOW HOMES 命令列出所有 ADR 主目录，然后使用 SET HOMEPATH 命令更改当前的 ADR 主目录。

然后，可以输入 SHOW HM_RUN 命令列出 ADR 资料档案库中注册的且在 V\$HM_RUN 中可以找到的所有检查器运行。找到要为其创建报表的检查器运行，使用相应 RUN_NAME 字段记下检查器运行名称。如果此检查器运行的报表已存在，则 REPORT_FILE 字段中将包含一个文件名。如果不存在，可以使用 CREATE REPORT HM_RUN 命令生成报表，如幻灯片中所显示。要查看报表，请使用 SHOW REPORT HM_RUN 命令。

测验

数据恢复指导可以处理两种情况：一种情况是您无法启动数据库（因为缺少一些必需的数据库文件，或者这些数据库文件不一致或已损坏），另一种情况是运行时发现文件损坏。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：1

测验

执行 ADVISE FAILURE 命令之后，将自动执行修复。因此，系统不再由您控制。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2

测验

ADR 驻留在数据库中。因此，必须装载实例才能进行意外事件分析。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2

测验

健康状况监视器可执行下列哪些检查？

1. 直观提交检查
2. 内存检查
3. 元数据检查
4. 重做检查
5. 事务处理检查
6. 用户预警检查
7. 还原段检查

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2、3、4、5、7

小结

在本课中，您应该已经学会：

- 检测和修复数据库损坏：
 - 使用新的 RMAN 数据修复命令执行以下操作：
 - 列出故障
 - 接收修复建议
 - 修复故障
 - 执行主动故障检查
- 处理块损坏：
 - 实时验证块完整性
 - 执行块介质恢复
- 设置自动诊断资料档案库
- 运行健康检查

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 9 概览：诊断数据库

本练习包含以下主题：

- 发现损坏
- 修复损坏

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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

10

使用闪回技术 I

ORACLE®

版权所有 © 2011, Oracle。保留所有权利。

课程目标

闪回
> - 概览
- 查询
- 表
- 事务处理

学完本课后，应能完成以下工作：

- 描述闪回技术
- 执行闪回查询
- 使用闪回版本查询
- 对表启用行移动
- 执行闪回表操作
- 使用闪回事务处理查询
- 使用闪回事务处理

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

闪回技术

对象级别	方案示例	闪回技术	依赖的对象	影响数据
数据库	截断表；发生了意外多表更改事件	数据库	闪回日志	是
表	删除表	删除	回收站	是
	使用了错误的 WHERE 子句进行更新	表	还原数据	是
	将当前数据与过去数据进行比较	查询	还原数据	否
	比较行版本	版本	还原数据	否
	保留历史事务处理数据	数据归档	还原数据	是
事务处理	调查并回退可疑事务处理	事务处理	来自归档日志的撤销/重做数据	是

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回技术

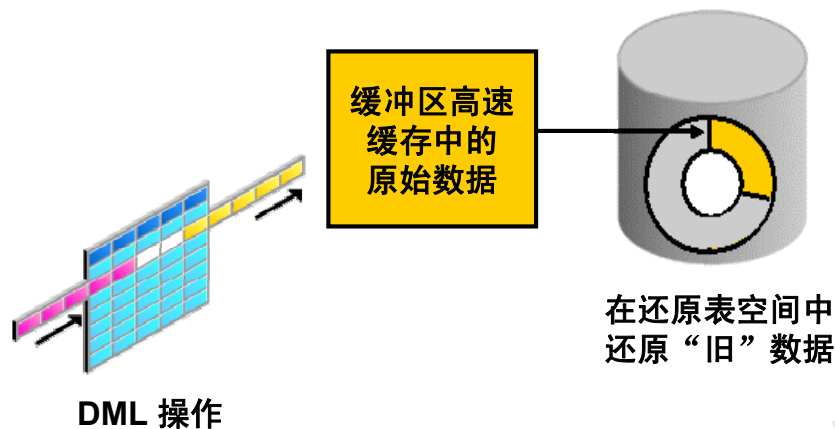
当 Oracle DB 发生逻辑损坏，而又需要简单快捷地恢复数据时，可以使用闪回技术。与人为错误一样，很难确定错误事务处理所影响的对象和行。通过使用闪回技术，可以诊断错误在数据库中是如何造成的，然后可以修复损坏。可以查看导致修改特定行的事务处理，查看特定时间段内给定行的所有版本集，或者只查看过去特定时间显示的数据。幻灯片表中显示了闪回技术的典型用法。闪回数据库依赖于闪回日志来执行闪回。闪回删除功能使用回收站，其它所有技术都使用还原数据。

并不是所有闪回功能都会修改数据库。有些功能只是用来查询数据的其它版本的方法；您可以使用这些工具来调查问题，并帮助进行恢复。闪回查询的结果可以帮助您执行以下两种操作之一：

- 确定要执行哪类数据库修改闪回操作来修复问题。
- 将这些查询的结果集输入到 INSERT、UPDATE 或 DELETE 语句中，便可以轻松地修复错误数据。

通过闪回数据归档，可使用前面的逻辑闪回功能访问过去很久以前的数据。

事务处理和还原



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

事务处理和还原

事务处理开始时，它被分配到一个还原段。在事务处理的整个过程中，更改数据时，原始“旧”值会复制到还原段。通过检查 V\$TRANSACTION 视图，可以查看哪些事务处理被分配到哪些还原段。

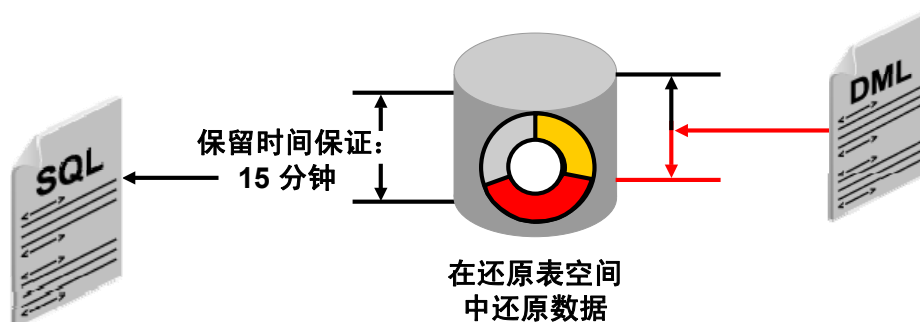
还原段是为了支持事务处理，根据需要由实例自动创建的专用段。像所有段一样，还原段由区组成，区又由数据块组成。还原段根据需要可自动增长和收缩，它充当一个用于所分配的事务处理的循环存储缓冲区。

当事务处理填满其当前还原段区中的块时，会为事务处理分配相同区中的其它块。如果该区中没有空闲块，则事务处理获取段中下一个区的块。占用了所有区之后，事务处理会自动转回到第一个区或请求为还原段分配新区。

幻灯片中的图表在左侧显示了一个表图标，以及由 DML 操作获得的原始数据。原始数据保留在缓冲区高速缓存中（如果未过期），然后写入到还原表空间（显示在右侧的圆环中）。

注：并行 DML 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》。

保证还原保留时间



运行时间不超过 15 分钟的
SELECT 语句始终
可以得到满意的结果。

生成的还原数据
多于可用空间时
事务处理会**失败**。

ORACLE

版权所有 © 2011, 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、尚未失效的事务处理，而不是因为缺少还原空间而使活动事务处理失败。出现冲突时，事务处理优先于查询。

此行为在保证了保留时间后会有所改变。有了保证的保留时间，即使会导致事务处理失败，仍然会强制执行还原保留时间设置。（因此，出现冲突时，查询优先于事务处理。）

RETENTION GUARANTEE 是表空间属性而不是初始化参数。可使用 SQL 命令行语句或 Oracle Enterprise Manager 更改此属性。通过更改还原表空间来保证保留时间的语法是：

```
SQL> ALTER TABLESPACE undotbs1 RETENTION GUARANTEE;
```

要将保证的还原表空间恢复至其常规设置，请使用以下命令：

```
SQL> ALTER TABLESPACE undotbs1 RETENTION NOGUARANTEE;
```

可以在 Oracle Enterprise Manager 中设置还原保留时间保证。导航到 “Automatic Undo Management（自动还原管理）” 页。单击保留时间保证的当前设置（“General/Undo Retention Settings（一般/还原保留时间设置）”）进行修改。

准备数据库以进行闪回

- 创建还原表空间
- 启用自动还原管理
- 指定还原保留时间和保证还原保留时间
- 默认的数据库初始化参数：
 - `UNDO_MANAGEMENT='AUTO'`
 - `UNDO_TABLESPACE='UNDOTBS1'`
 - `UNDO_RETENTION=900`

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

准备数据库以进行闪回

要启用应用程序的闪回功能，必须执行以下任务：

- 创建具有足够空间的还原表空间以保留闪回操作所需的数据。用户更新数据越频繁，所需空间就越大。数据库管理员通常会计算空间要求。如果您不确定空间要求，则可以先创建一个可自动扩展的还原表空间，观察该表空间一个业务周期（例如，1 或 2 天），使用 `V$UNDO_STAT` 视图收集还原块信息，计算空间要求，然后使用这些空间要求来创建大小合适的固定还原表空间。（《Oracle 数据库管理员指南》中提供了计算公式。）
- 默认情况下会启用自动还原管理。如果需要，请按《Oracle 数据库管理员指南》中的说明启用自动还原管理。
- 对于固定大小的还原表空间，Oracle DB 会自动优化系统，以便为还原表空间提供可能的最佳还原保留时间。
- 对于可自动扩展的还原表空间（默认），Oracle DB 会保留还原数据，其保留时间至少满足运行时间最长的查询所需的保留期，以及 `UNDO_RETENTION` 参数指定的还原保留时间阈值。

准备数据库以进行闪回（续）

可以查询 `V$UNDOSTAT.TUNED_UNDORETENTION` 来确定对当前还原表空间的还原数据的保留时间。设置 `UNDO_RETENTION` 参数并不能保证未失效的还原数据不会被覆盖。如果系统需要更多空间，Oracle DB 可以用最新生成的还原数据覆盖未失效的还原数据。

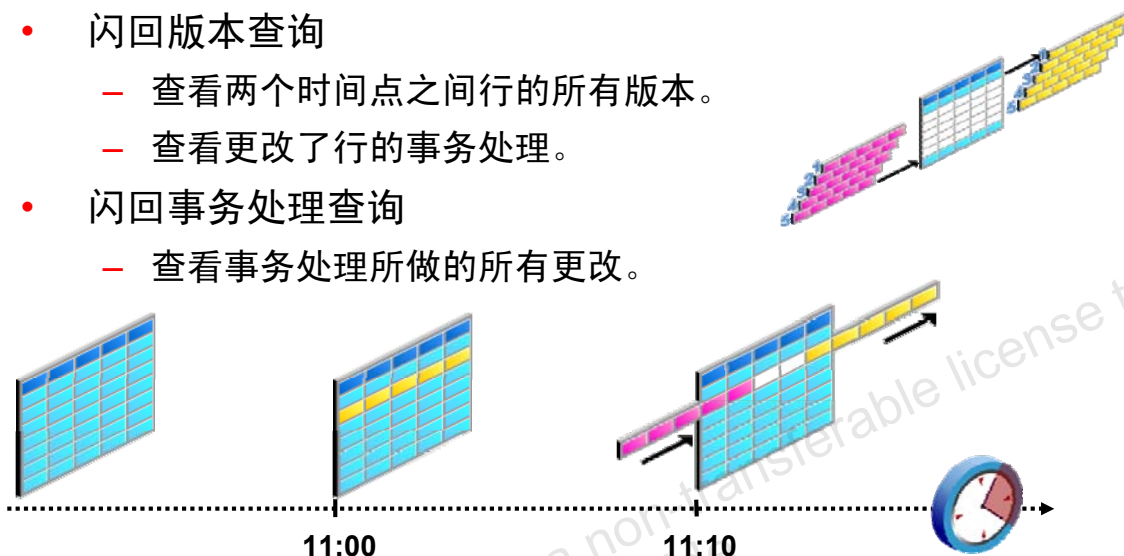
- 为还原表空间指定 `RETENTION GUARANTEE` 子句，确保不会丢弃未失效的还原数据。
- 为需要使用闪回功能的用户、角色或应用程序授予闪回权限。

要满足长期保留要求，请创建闪回数据归档。

使用闪回技术查询数据

闪回
- 概览
> - 查询
- 表
- 事务处理

- 闪回查询
 - 查询指定时间点的所有数据。
- 闪回版本查询
 - 查看两个时间点之间行的所有版本。
 - 查看更改了行的事务处理。
- 闪回事务处理查询
 - 查看事务处理所做的所有更改。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

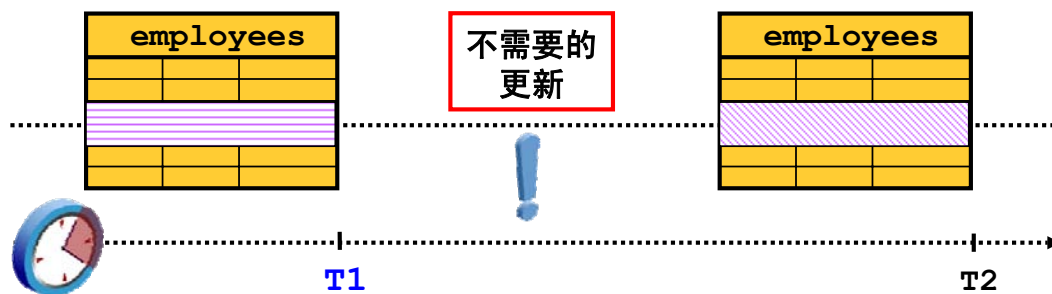
使用闪回技术查询数据

闪回技术提供了查询以前版本的方案对象、查询历史记录数据和执行更改分析的功能。每个事务处理在逻辑上都会生成一个新版本的数据库。借助闪回技术，可通过浏览这些版本来查找错误及起因：

- **闪回查询：**查询在特定时间点存在的所有数据。
- **闪回版本查询：**查看两个时间点之间行的所有版本以及更改了行的事务处理。
- **闪回事务处理查询：**查看某个事务处理所做的所有更改，并且，如果需要，使用“还原”SQL 命令回退事务处理。

闪回查询

用于查询指定时间点的所有数据。



```
SELECT employee_id, salary FROM employees
AS OF TIMESTAMP <T1>
WHERE employee_id = 200
```

ORACLE

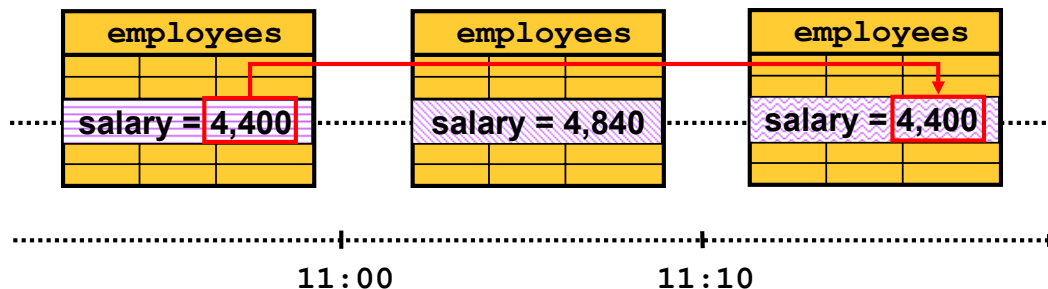
版权所有 © 2011, Oracle。保留所有权利。

闪回查询

使用闪回查询功能，可以执行截止到特定时间的查询。使用 SELECT 语句的 AS OF 子句，可以指定要查看其对应数据的时间戳。这在分析数据差异时非常有用。

注：TIMESTAMP 和 SCN 是 AS OF 子句的有效选项。

闪回查询：示例



```

UPDATE employees
SET salary =
  (SELECT salary FROM employees
   AS OF TIMESTAMP TO_TIMESTAMP
    ('2005-05-04 11:00:00', 'yyyy-mm-dd hh24:mi:ss')
   WHERE employee_id = 200)
WHERE employee_id = 200
  
```

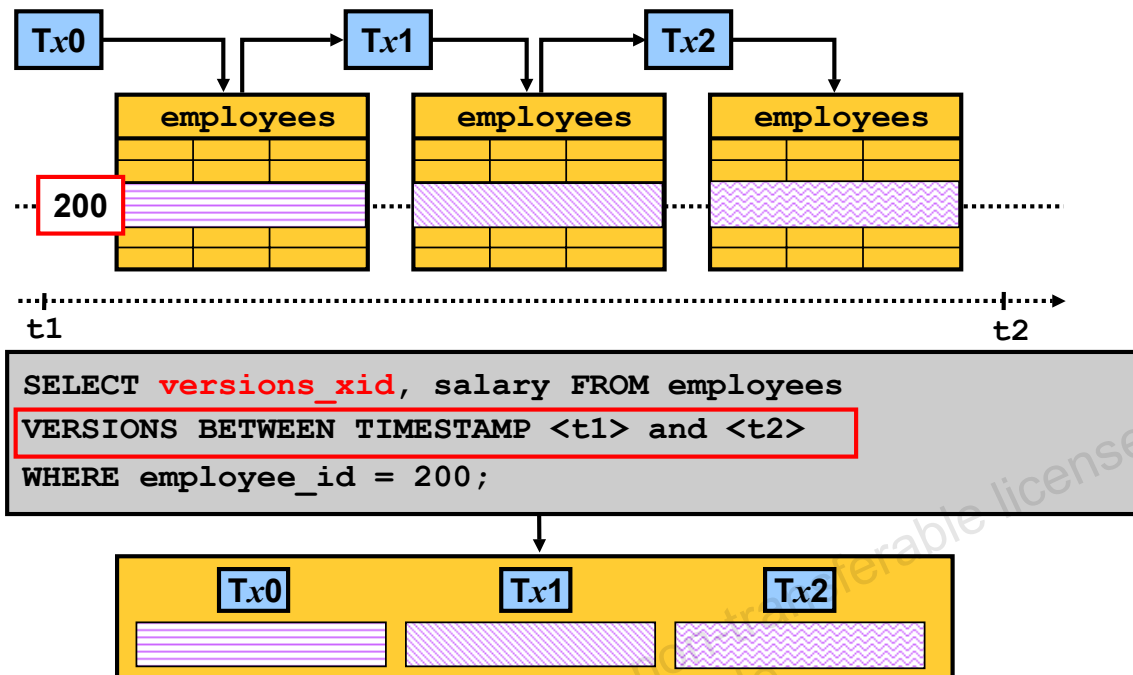
ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回查询：示例

如果最近错误地给某个雇员加了薪，则可重新更改薪金，其值由子查询返回的闪回值提供。

闪回版本查询



版权所有 © 2011, Oracle. 保留所有权利。

闪回版本查询

使用闪回查询，可对特定时间范围或用户指定的系统更改号 (SCN) 范围内的数据库进行查询。通过闪回版本查询功能，可使用 VERSIONS 子句检索两个时间点或两个 SCN 之间的所有行的所有版本。

闪回版本查询返回的那些行表示跨事务处理的那些行的更改历史记录。闪回版本查询只检索提交的行。不显示事务处理中未提交的行版本。返回的行还包括行的已删除版本和后来重新插入的版本。

可以使用闪回版本查询检索行历史记录。闪回版本查询提供了一种方法，通过这种方法可审计表行，检索影响行的事务处理的有关信息。然后，可以使用返回的事务处理标识符来执行事务处理挖掘（通过使用 LogMiner）或执行闪回事务处理查询，本课稍后会对此进行论述。

注：VERSIONS_XID 是返回相应行版本的事务处理标识符的伪列。

闪回版本查询：注意事项

- 不能使用 VERSIONS 子句查询：
 - 外部表
 - 临时表
 - 固定表
 - 视图
- VERSIONS 子句不能跨 DDL 命令使用。
- 段收缩操作已过滤掉。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回版本查询：注意事项

VERSIONS 子句不能用于查询下列类型的表：

- 外部表
- 临时表
- 固定表

不能使用 VERSIONS 子句查询视图。但是，在视图定义中可使用 VERSIONS 子句。

SELECT 语句中的 VERSIONS 子句不能跨更改相应表的结构的多条 DDL 语句生成行的版本。这意味着查询到达某个过去时间（此时更改了表结构）后会停止生成行。

诸如段收缩这样的维护操作可能会跨块移动表行。在这种情况下，因为行数据保持不变，所以版本查询会过滤掉此类幻像版本。

测验

闪回查询将当前数据与过去数据进行比较。为此，它同时使用还原和重做数据。

1. 对
2. 错

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：2

测验

选择正确的语句：

1. 闪回版本查询使用还原数据并修改数据。
2. 闪回版本查询使用还原数据，不修改数据。
3. 闪回版本查询同时使用还原和重做数据。

ORACLE

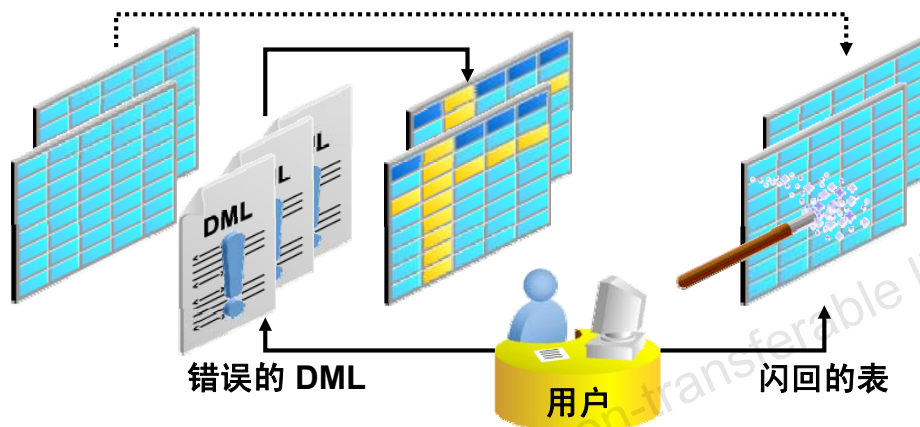
版权所有 © 2011, Oracle。保留所有权利。

答案：2

闪回表概览

闪回
- 概览
- 查询
- 表
- 事务处理

- 闪回表可将表恢复到特定的时间点。
- 闪回表是一个原位操作。
- 数据库保持联机。



ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

闪回表概览

使用闪回表功能，可将一组表恢复到特定的时间点，而不需要执行传统的时间点恢复操作。

闪回表操作是在数据库处于联机状态时，通过只回退对指定表及其从属对象所做的更改，原位完成的。

闪回表语句可作为单个事务处理来执行。必须成功地闪回所有表，否则会回退整个事务处理。

注：可以使用闪回版本查询和闪回事务处理查询来确定适当的闪回时间。

闪回表

- 使用闪回表，可将一个或多个表还原到特定时间点，而不需要还原备份。
- 将从还原表空间检索数据来执行闪回表操作。
- 您需要具有对特定表的 FLASHBACK ANY TABLE 或 FLASHBACK 对象权限。
- 需要对要闪回的表具有 SELECT、INSERT、DELETE 和 ALTER 权限。
- 必须对要执行闪回操作的表启用行移动。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回表

使用闪回表时，可以将一个或多个表还原到特定时间点，而不需要还原备份。使用此功能时，会还原表及其关联对象（如索引、约束条件、触发器等）中的数据。可从还原表空间检索用于满足闪回表请求的数据。可以使用闪回版本查询和闪回事务处理查询确定适当的闪回时间。

闪回表提供了一种方法，通过这种方法用户可方便快速地恢复意外的修改，而不需要数据库管理员的干预。必须为使用闪回表功能的所有用户授予 FLASHBACK TABLE 或 FLASHBACK ANY TABLE 系统权限。此外，还必须为用户授予 SELECT、INSERT、DELETE 和 ALTER 对象权限。

可以使用 Oracle Enterprise Manager 对表执行闪回操作。向导会引导您完成整个过程。

注：下一页介绍了如何启用行移动。

对表启用行移动

Edit Table: HR.EMPLOYEES

Actions: Create Like [Go] [Show SQL] [Revert] [Apply]

General Constraints Segments Storage **Options** Statistics Indexes

Enable Row Movement: Yes

☐ Parallel - Use multiple threads when creating this object or when executing DML against this object.
Parallel Degree: ☒ Default ☐ Value []

☐ Cache - Place frequently accessed data to the top of the buffer cache.

General Constraints Segments Storage **Options** Statistics Indexes

```
ALTER TABLE employees ENABLE ROW MOVEMENT;
```

ORACLE

版权所有 © 2011, Oracle. 保留所有权利。

对表启用行移动

必须先对表启用行移动，然后才能对表执行闪回操作。启用了行移动之后，Oracle Server 便可移动表中的行。

使用 Oracle Enterprise Manager 时，可通过执行以下步骤对表启用行移动：

1. 在“Schema（方案）”属性页的“Database Objects（数据库对象）”区域中，选择“Tables（表）”。输入方案名以搜索表，然后单击“Go（开始）”。
2. 单击要启用行移动的表的名称。此时会出现“View Table（查看表）”页。
3. 单击“Edit（编辑）”进入“Edit Table（编辑表）”页。
4. 单击“Options（选项）”选项卡，可在其中更改表的“Enable Row Movement（启用行移动）”设置。
5. 将“Enable Row Movement（启用行移动）”设置为“Yes（是）”，然后单击“Apply（应用）”。此时会显示更新确认消息。

执行闪回表

Perform Object Level Recovery: Point-in-time

Recovery Scope **Tables** Cancel Step 1 of 7 Next

Operation Type **Flashback Existing Tables**

Specify the point in time to which to recover.

☒ Evaluate row changes and transactions to decide on a point in time

Table Example: SCOTT.EMP

☐ Flashback to a timestamp

Date Time ☒ AM ☐ PM Example: Mar 19, 2003

☐ Flashback to a restore point

Restore Point

☐ Flashback to a known SCN

SCN

Perform Recovery

Oracle Advised Recovery

Oracle did not detect any failures. Advise and Recover

User Directed Recovery

Recovery Scope Recover

Operation Type ☒ Flashback Existing Tables ☐ Flashback Dropped Tables

FLASHBACK TABLE hr.departments TO TIMESTAMP TO_TIMESTAMP('2007-04-05 21:00:00', 'YYYY-MM-DD HH24:MI:SS');

版权所有 © 2011, Oracle. 保留所有权利。

执行闪回表

可以使用 Oracle Enterprise Manager 对表执行闪回操作，步骤如下：

1. 在“Availability（可用性）”属性页上的“Backup/Recovery（备份/恢复）”区域中，选择“Perform Recovery（执行恢复）”。
2. 在“Object Level Recovery（对象层次恢复）”区域中，从“Object Type（对象类型）”下拉列表选择“Tables（表）”。
3. 选择“Flashback existing tables（闪回现有表）”作为“Operation Type（操作类型）”。单击“Recover（恢复）”。此时会显示“Perform Object Level Recovery: Point-in-time（执行对象级恢复: 时间点）”页。
4. 选择“Flashback to a timestamp（闪回至时间戳）”或“Flashback to a known SCN（闪回至已知 SCN）”，指定要闪回到的时间戳或 SCN，然后单击“Next（下一步）”。
5. 单击“Add Tables（添加表）”，将表添加到闪回操作列表。单击“Next（下一步）”。
6. 如果存在从属表，则会显示“Dependency Options（相关性选项）”页。选择处理从属表的所需选项。通常，应选择“Cascade（级联）”来确保闪回一致。单击“Next（下一步）”。
7. 此时会显示“Perform Object Level Recovery: Review（执行对象级恢复: 复查）”页。复查信息后单击“Submit（提交）”。此时会出现“Confirmation（确认）”页。

执行闪回表（续）

注：还可以使用“Administration（管理）”页的“Schema（方案）”区域中的“Tables（表）”链接来闪回表。

闪回表：注意事项

- FLASHBACK TABLE 命令作为单个事务处理执行，要求获取 DML 排它锁。
- 不闪回统计信息。
- 保留当前索引和从属对象。
- 闪回表操作：
 - 不能对系统表执行
 - 不能跨 DDL 操作
 - 会生成还原和重做数据

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回表：注意事项

- 整个 FLASHBACK TABLE 语句是在一个事务处理中执行的。将闪回所有指定表，或者不闪回任何指定表。
- 在执行操作的整个过程中，闪回表会获取对该语句指定的所有表的数据操纵语言 (DML) 排它锁。
- 不会闪回受影响对象的统计信息。
- 会保留所有现有索引。不会重新创建删除的索引。还会自动保留相关的提交时实体化视图。
- 只要不违反任何表约束条件，就会闪回 FLASHBACK TABLE 语句中指定的表。如果在闪回执行期间违反了任何约束条件，则会中止操作，此时表的状态与调用 FLASHBACK TABLE 语句之前的状态相同。
- 不能将表闪回至早于执行数据定义语言 (DDL) 操作时的某个特定时间，因为执行数据定义语言 (DDL) 操作变更了闪回操作中涉及的表的结构或收缩了表。只更改表的存储属性的 DDL 语句不受此限制。
- 不能对系统表、远程表和固定表执行闪回表操作。

测验

选择所有正确的语句：

1. 在闪回表时，数据库可保持打开状态。
2. 闪回表是作为单个事务处理执行的。
3. 闪回表需要可用的备份。
4. 闪回表基于还原数据。

ORACLE

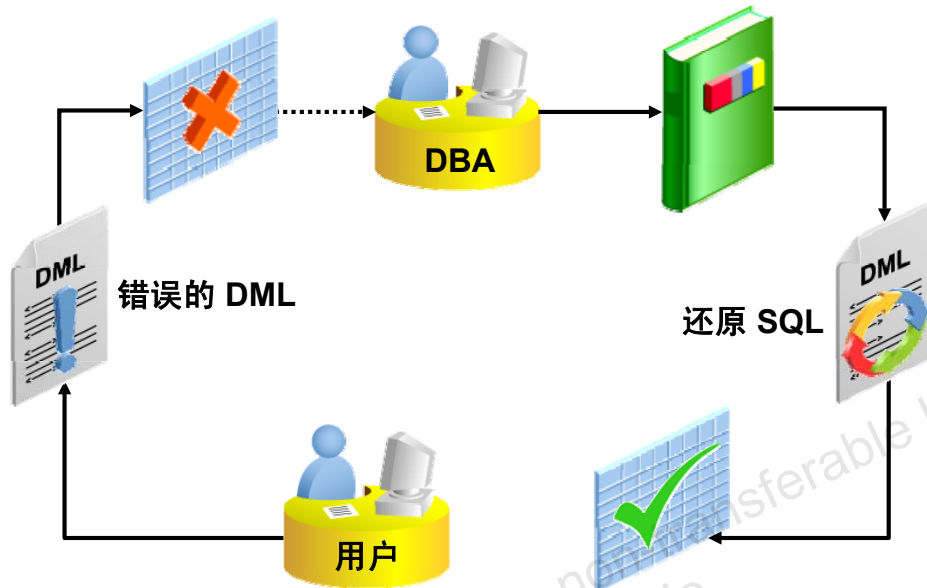
版权所有 © 2011, Oracle。保留所有权利。

答案：1、2、4

闪回事务处理查询

- 闪回
- 概览
- 查询
- 表
- > - 事务处理

FLASHBACK_TRANSACTION_QUERY



ORACLE

版权所有 © 2011, Oracle。保留所有权利。


闪回事务处理查询

闪回事务处理查询是一种用于查看在事务处理级对数据库所做更改的诊断工具。这样，可诊断数据库中的问题并对事务处理执行分析和审计。

可以使用 FLASHBACK_TRANSACTION_QUERY 视图来确定所有必要的 SQL 语句，以便用来还原由特定事务处理所做的或在特定时间段内所做的修改。

使用 Oracle Enterprise Manager 执行闪回事务处理查询

Select	Transaction ID	DB User	Commit Timestamp	Redo Records	Transaction Summary - Updates (upd), Inserts (ins), Deletes (del), Other (oth)
	06001600C2030000	HR	Aug 20, 2009 4:17:03 PM	5	HR. REGIONS (5 ins)
	08001C0052030000	HR	Aug 20, 2009 4:17:05 PM	2	HR. REGIONS (2 upd)
	040010000E030000	HR	Aug 20, 2009 4:17:06 PM	3	HR. REGIONS (3 upd)
	09001F00A0030000	HR	Aug 20, 2009 4:17:07 PM	1	HR. REGIONS (1 upd)

Transaction ID 08001C0052030000		Start SCN 1106834	Start Time Aug 20, 2009 4:17:05 PM	
DB User UNKNOWN		Commit SCN 1106835	Commit Time Aug 20, 2009 4:17:05 PM	
OS User		Machine Name		
SCN 	Operation	Schema	Table	SQL Redo
1106834	START			set transaction read write;
1106834	UPDATE	HR	REGIONS	update "HR"."REGIONS" set "REGION_NAME" = 'Two Poles' where "REGION_ID" = 10 and "REGION_NAME" = 'Pole' and ROWID = 'AAAR5JAAFAAAACNAAA';
1106834	UPDATE	HR	REGIONS	update "HR"."REGIONS" set "REGION_NAME" = 'Many Moons' where "REGION_ID" = 20 and "REGION_NAME" = 'Moon' and ROWID = 'AAAR5JAAFAAAACNAAB';
1106835	COMMIT			commit;

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

使用 Oracle Enterprise Manager 执行闪回事务处理查询

借助执行恢复向导，这个功能可与闪回版本查询功能结合使用。在“Perform Object Level Recovery: Choose SCN（执行对象级恢复: 选择 SCN）”页上，单击“Flashback Version Query Result（闪回版本查询结果）”部分中对应的“Transaction ID（事务处理 ID）”链接。

在幻灯片示例中，对 JOBS 表执行闪回版本查询后，检索到 JOB_ID = 'AD_PRES' 的 JOBS 行有三个版本。然后，单击其中一个事务处理 ID，显示属于该事务处理的所有更改。注意，除了更新 JOBS 表之外，在该事务处理中还更新了 EMPLOYEES 表。

闪回事务处理查询：注意事项

- DDL 命令被视为字典更新。
- 通过执行 DDL 命令对事务处理执行闪回事务处理查询时，会显示对数据字典所做的更改。
- 删除的对象显示为对象编号。
- 删除的用户显示为用户标识符。

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回事务处理查询：注意事项

在数据库中，DDL 操作只是一系列空间管理操作和对数据字典所做的更改。通过执行 DDL 命令对事务处理执行闪回事务处理查询时，会显示对数据字典所做的更改。

当闪回事务处理查询涉及从数据库中删除的表时，不会反映表名，而是使用对象编号。

如果删除了执行事务处理的用户，则该事务处理的闪回事务处理查询只显示相应的用户 ID，而不是用户名。

注：当特定的事务处理没有足够的还原数据时，在 FLASHBACK_TRANSACTION_QUERY 的 OPERATION 列中会返回值为 UNKNOWN 的行。

闪回事务处理

- 建立闪回事务处理的先决条件
- 逐步完成可能的工作流
- 使用闪回事务处理向导
- 查询有依赖关系及没有依赖关系的事务处理
- 选择回退选项和闪回事务处理
- 检查结果

ORACLE

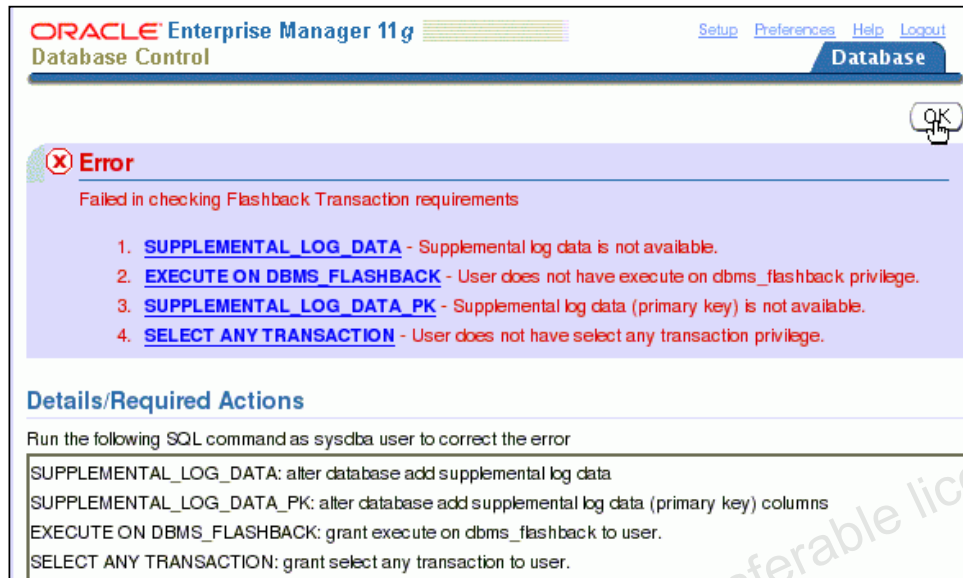
版权所有 © 2011, Oracle。保留所有权利。

闪回事务处理

使用闪回事务处理，可以撤消事务处理及从属事务处理。Oracle DB 将确定事务处理之间的依赖关系，实际上，它会创建一个撤消不需要的更改的校正事务处理。数据库会倒回到似乎从未发生过该事务处理及从属于它的任何事务处理的状态。

可从 Oracle Enterprise Manager 或 PL/SQL 程序包中使用闪回事务处理功能。

先决条件



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

先决条件

要使用此功能，必须启用补充日志记录功能并建立正确的权限。例如，HR 方案中的 HR 用户决定对 REGIONS 表使用闪回事务处理。SYSDBA 在 SQL*Plus 中执行下列设置步骤：

```
alter database add supplemental log data;
alter database add supplemental log data (primary key) columns;
grant execute on dbms_flashback to hr;
grant select any transaction to hr;
```

闪回事务处理

- 可使用 Oracle Enterprise Manager 或命令行闪回事务处理。
- EM 使用闪回事务处理向导，该向导调用带 NOCASCADE 选项的 `DBMS_FLASHBACK.TRANSACTION_BACKOUT` 过程。
- 如果 PL/SQL 调用成功完成，即意味着事务处理不具有任何依赖关系，并且已成功回退单个事务处理。



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回事务处理

安全权限

要闪回或回退事务处理，也就是创建校正事务处理，必须对所有受影响的表具有 SELECT、FLASHBACK 和 DML 权限。

使用条件

- 不支持跨相冲突的 DDL 的事务处理回退。
- 事务处理回退从 LogMiner 中继承数据类型支持。要查看支持的数据类型，请参阅 Oracle Database 11g 文档。

建议

- 发现需要回退事务处理时，尽快启动回退操作的效果会比较好。重做日志太大和事务处理率过高会降低事务处理回退操作的速度。
- 为回退操作提供事务处理名称有利于稍后的审计工作。如果没有提供事务处理名称，会自动生成一个名称。

可能的工作流

1. 查看表中的数据
2. 发现逻辑问题
3. 使用闪回事务处理
 1. 执行查询
 2. 选择事务处理
 3. 闪回事务处理（如果没有冲突）
 4. 选择其它回退选项（如果存在冲突）
4. 查看闪回事务处理结果



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

可能的工作流

假定有如下所示的几个事务处理：

```
connect hr
Enter password: oracle_4U <<< not displayed
INSERT INTO hr.regions VALUES (5, 'Pole');
COMMIT;
UPDATE hr.regions SET region_name='Poles' WHERE region_id = 5;
UPDATE hr.regions SET region_name='North and South Poles' WHERE
region_id = 5;
COMMIT;
INSERT INTO hr.countries VALUES ('TT', 'Test Country', 5);
COMMIT;
connect sys/<password> as sysdba
ALTER SYSTEM ARCHIVE LOG CURRENT;
```


闪回事务处理向导

Flashback Transaction: Perform Query

Database: **orcl** Operation Type: **Flashback Transaction** Cancel Step 1 of 4 Next

Specify the time range to begin querying. The time range is initialized to the last hour but any available (on disk) archived logs can be queried. You may need to specify additional columns (or reduce the time range) to further narrow the results.

Query Time Range

☒ Time Range ☐ SCN Range

* Start Time: Aug 20, 2009 05:19 AM ☒ PM

* End Time: Aug 20, 2009 05:33 PM ☒ PM

☒ TIP The oldest time available on disk is Aug 20, 2009 5:19:26 PM [View Archived Logs](#)

☒ TIP Earlier start times are available by [restoring archived logs](#)

Query Filter

Table: **HR.REGIONS**

Examples: Scott.Emp, Scott.%

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

闪回事务处理向导

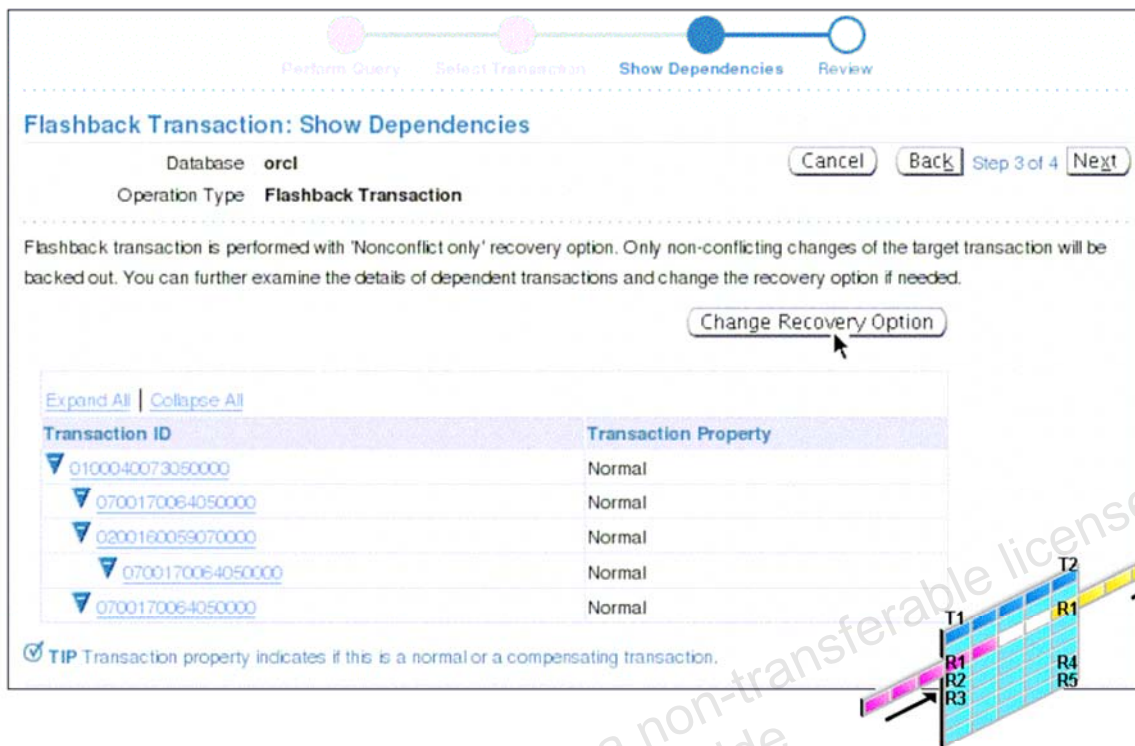
在 Oracle Enterprise Manager 中，在“Table（表）”下方选择 HR.REGIONS，从“Actions（操作）”下拉列表中选择“Flashback Transaction（闪回事务处理）”，然后单击“Go（开始）”。此时会针对选定表调用闪回事务处理向导，并显示“Flashback Transaction: Perform Query（闪回事务处理: 执行查询）”页。

选择适当的时间范围，然后添加查询参数（指定的范围越具体，闪回事务处理向导进行搜索所用的时间就越短）。

如果没有 Oracle Enterprise Manager，请使用 DBMS_FLASHBACK.TRANSACTION_BACKOUT 过程（《Oracle Database PL/SQL Packages and Types Reference》中提供了有关此过程的说明）。实际上，您可以将事务处理 ID 数组作为依赖关系搜索的起点。例如：

```
CREATE TYPE XID_ARRAY AS VARRAY(100) OF RAW(8);
CREATE OR REPLACE PROCEDURE TRANSACTION_BACKOUT(
  numberOfXIDs NUMBER, -- number of transactions passed as input
  xids XID_ARRAY, -- the list of transaction ids
  options NUMBER default NOCASCADE, -- back out dependent
  txn timeHint TIMESTAMP default MINTIME -- time hint on the txn
  start
);
```

选择其它回退选项



版权所有 © 2011, Oracle. 保留所有权利。

选择其它回退选项

TRANSACTION_BACKOUT 过程会检查依赖关系，例如：

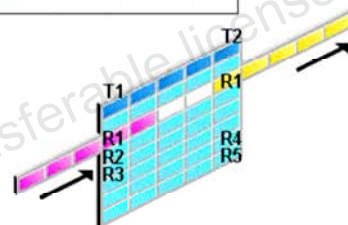
- 连续编写 (WAW)
- 主约束条件和唯一约束条件
- 外键约束条件

事务处理可以具有 WAW 依赖关系，这意味着事务处理将更新或删除由从属事务处理插入或更新的行。例如，在主（或唯一）约束条件和强制外键约束条件的主/从关系中会发生这种情况。

要了解 NONCONFLICT_ONLY 和 NOCASCADE_FORCE 选项之间的差异，可以假定 T1 事务处理更改了行 R1、R2 和 R3，T2 事务处理更改了行 R1、R3 和 R4。在此方案中，因为两个事务处理都更新了 R1 行，因此 R1 行成了“冲突”行。T2 事务处理与 T1 事务处理具有 WAW 依赖关系。使用 NONCONFLICT_ONLY 选项，可回退 R2 和 R3，因为 R2 和 R3 不存在冲突并且假定您知道处理 R1 行的方法。使用 NOCASCADE_FORCE 选项，可回退全部三个行（R1、R2 和 R3）。

注：此屏幕快照不是工作流示例的一部分，但它显示了较复杂情况的其它详细资料。

选择其它回退选项



ORACLE

版权所有 © 2011, Oracle。保留所有权利。

选择其它回退选项（续）

闪回事务处理向导按以下方式运行：

如果带有 NOCASCADE 选项的 DBMS_FLASHBACK.TRANSACTION_BACKOUT 过程失败（因为存在从属事务处理），则可以更改恢复选项。

- 使用“Nonconflict Only（仅限无冲突）”选项，可回退事务处理中的非冲突行，从而保持了数据库的一致性（虽然为了修复数据而破坏了事务处理的原子性）。
- 如果希望不考虑从属事务处理而强制回退给定的事务处理，请使用“Nocascade Force（强制无级联）”选项。服务器按照与提交时间相反的顺序仅对给定事务处理执行校正 DML 命令。如果没有违反约束条件，您可以继续提交更改，否则会回退。
- 要按投寄顺序完全删除给定事务处理及其所有从属事务处理，请使用“Cascade（级联）”选项。

注：此屏幕快照不是工作流示例的一部分，但它显示了较复杂情况的其它详细资料。

最后步骤（不使用 EM）

选择回退选项后，会在 DBA_FLASHBACK_TXN_STATE 和 DBA_FLASHBACK_TXN_REPORT 视图中生成相关性报告。

- 检查显示了已回退的所有事务处理的相关性报告。
- 提交更改使其成为永久更改。
- 回退以放弃更改。



版权所有 © 2011, Oracle。保留所有权利。

最后步骤（不使用 EM）

DBA_FLASHBACK_TXN_STATE 视图包含事务处理的当前状态：该事务处理在系统中处于活动状态还是已被有效回退。系统将使用校正事务处理以原子方式维护此表。对于每个校正事务处理，可能存在多个行，其中每行都会提供校正事务处理已校正的事务处理间的依赖关系。

DBA_FLASHBACK_TXN_REPORT 视图可提供关于数据库中已提交的所有校正事务处理的详细信息。此视图中的每一行都与一个校正事务处理相关联。

有关这些表的详细说明，请参阅《Oracle Database Reference》。

测验

您发现 Jim 的薪金更新了两次。第一次更新是正确的，但第二次是由失误造成的。在您发现该问题之前，EMPLOYEES 表中的其它行都已正确更新。您应使用哪种技术来修复该错误？

1. 闪回数据库
2. 闪回查询
3. 闪回事务处理

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

答案：3

小结

在本课中，您应该已经学会：

- 描述闪回技术
- 执行闪回查询
- 使用闪回版本查询
- 对表启用行移动
- 执行闪回表操作
- 使用闪回事务处理查询
- 使用闪回事务处理

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

练习 10 概览：执行闪回事务处理回退

本练习包含以下主题：

- 查询事务处理
- 执行闪回事务处理回退

ORACLE

版权所有 © 2011, Oracle。保留所有权利。

Unauthorized reproduction or distribution prohibited. Copyright© 2014, Oracle and/or its affiliates.

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