DG学习

目录

[1. Data Guard 容灾机制 4](#_Toc29240)

[1.1. 概述 5](#_Toc1626)

[1.1.1. 主要功能包括： 5](#_Toc12473)

[1.2. Standby Database（备库）的类型 5](#_Toc30844)

[1.2.1. Physical Standby Database: 物理备库 5](#_Toc6911)

[1.2.2. Logical Standby Database: 逻辑备库 5](#_Toc6845)

[1.2.3. Snapshot Standby Database: 快照备库 5](#_Toc10141)

[1.3. 了解两种redo transport modes 5](#_Toc22244)

[1.3.1. 同步（Synchronous）重做传输模式： 5](#_Toc6351)

[1.3.2. 异步（Asynchronous）重做传输模式： 6](#_Toc12160)

[1.4. 相关参数 6](#_Toc13061)

[1.4.1. DB\_NAME： 6](#_Toc31451)

[1.4.2. DB\_UNIQUE\_NAME： 6](#_Toc28821)

[1.4.3. LOG\_ARCHIVE\_CONFIG： 6](#_Toc6754)

[1.4.4. CONTROL\_FILES： 6](#_Toc25600)

[1.4.5. LOG\_ARCHIVE\_DEST\_n： 6](#_Toc14919)

[1.4.6. LOG\_ARCHIVE\_DEST\_STATE\_n： 6](#_Toc17786)

[1.4.7. REMOTE\_LOGIN\_PASSWORDFILE： 6](#_Toc17277)

[1.4.8. LOG\_ARCHIVE\_FORMAT： 6](#_Toc19855)

[1.4.9. LOG\_ARCHIVE\_MAX\_PROCESSES： 6](#_Toc14901)

[1.4.10. FAL\_SERVER： 6](#_Toc19999)

[1.4.11. DB\_FILE\_NAME\_CONVERT： 7](#_Toc21884)

[1.4.12. LOG\_FILE\_NAME\_CONVERT： 7](#_Toc17415)

[1.4.13. STANDBY\_FILE\_MANAGEMENT： 7](#_Toc16566)

[1.5. 什么是ADG 7](#_Toc12071)

[1.5.1. ADG的主要特点包括： 7](#_Toc15880)

[1.6. 什么是MAA 7](#_Toc11190)

[1.6.1. MAA的主要组件包括： 7](#_Toc22764)

[1.7. 日志 8](#_Toc24311)

[1.7.1. 日志存储 8](#_Toc4739)

[1.7.2. 日志传送 8](#_Toc6857)

[1.7.3. 日志接收 8](#_Toc32100)

[1.8. 日志应用 8](#_Toc30393)

[1.8.1. Redo Apply-物理 DG 8](#_Toc11513)

[1.8.2. SQL Apply-逻辑 DG 8](#_Toc21101)

[1.9. 如何监控DG运行状态 9](#_Toc15526)

[1.9.1. 相关视图及命令 9](#_Toc5287)

[1.9.2. 主库环境检查 9](#_Toc5459)

[主库DG环境的保护模式检查 9](#_Toc21301)

[1.10. GAP 9](#_Toc3083)

[1.10.1. 什么是GAP 9](#_Toc13190)

[1.10.2. 如何处理GAP 9](#_Toc2766)

[1.11. 常见报错处理 10](#_Toc4284)

[1.11.1. ORA-10458: standby database requires recovery。ORA-01196: ORA-01110: 10](#_Toc15567)

[2. 创建物理 DG 11](#_Toc7024)

[2.1. 需求规划 11](#_Toc16812)

[2.2. 主库开归档 11](#_Toc2187)

[2.3. FORCE LOGGING 12](#_Toc18279)

[2.4. 创建主库 pfile 文件 12](#_Toc20439)

[2.4.1. 修改 pfile 文件 13](#_Toc10538)

[2.4.2. pfile 起库测试 14](#_Toc16445)

[2.4.3. 创建 spfile,重新启库 14](#_Toc24853)

[2.5. 备库参数文件、密码文件 15](#_Toc7879)

[2.5.1. 拷贝参数文件、密码文件到备库 15](#_Toc19889)

[2.5.2. 修改备库参数文件 15](#_Toc28385)

[2.5.3. 备库创建目录 16](#_Toc4025)

[2.5.4. 备库启动实例到 nomount 查看参数确认 16](#_Toc13675)

[2.5.5. 备库创建 spfile 重启实例到 nomount 17](#_Toc11088)

[2.6. 配置主、备库网络 17](#_Toc21919)

[2.6.1. 主库静态监听 17](#_Toc12266)

[(SID\_NAME = prod1) 18](#_Toc24019)

[2.7. 备库静态 listener 18](#_Toc1130)

[(SID\_NAME = test1) 18](#_Toc2975)

[2.7.1. 主备库 tnsnames.ora 19](#_Toc8517)

[2.7.2. 测试网络 19](#_Toc18932)

[2.8. rman duplicate 创建物理备库 20](#_Toc4268)

[2.8.1. 主备库打开告警日志 20](#_Toc16678)

[2.8.2. rman 登录主备库 20](#_Toc3333)

[2.8.3. 执行 rman duplicate 20](#_Toc4081)

[2.8.4. 查看主备库状态 20](#_Toc6210)

[2.8.5. open 打开备库 21](#_Toc15874)

[2.9. 备库启动 MRP 进程 应用日志 21](#_Toc31445)

[2.10. 测试 21](#_Toc15257)

[2.10.1. 主库切归档 测试 21](#_Toc20299)

[2.10.2. scott 建表插数据测试 22](#_Toc7977)

[2.11. 报错 24](#_Toc10210)

[2.11.1. 报错1 24](#_Toc6448)

[2.11.2. 报错2 24](#_Toc1418)

[2.11.3. 报错3 24](#_Toc6045)

[3. 快照DG 25](#_Toc21555)

[3.1. 设置 Fast recovery area 25](#_Toc7538)

[3.2. 备库取消日志应用 25](#_Toc32350)

[3.3. 备库关库启动到 mount 状态 25](#_Toc14482)

[3.4. 物理备库切 snapshot standby 25](#_Toc31293)

[3.5. 快照备库写测试 26](#_Toc3797)

[3.6. 主库插数据切日志测试 26](#_Toc1268)

[3.7. 快照 DG 切回物理 DG 26](#_Toc1235)

[3.8. 关闭备库重新启动 26](#_Toc9394)

[SYS@test1>startup 26](#_Toc21519)

[3.9. 备库应用日志，查看 26](#_Toc4283)

[4. 三种保护模式 27](#_Toc21346)

[4.1. 保护模式 27](#_Toc22171)

[4.1.1. 最大保护 maximum protection 27](#_Toc3032)

[4.1.2. 最高可用 maximum availability 27](#_Toc3608)

[4.1.3. 最大性能 (默认) maximum performance 28](#_Toc17758)

[4.2. 相关命令 28](#_Toc14758)

[4.2.1. 查看当前保护模式 28](#_Toc20977)

[4.3. 创建 standby redolog 28](#_Toc8028)

[4.3.1. 主库添加 standby redolog 29](#_Toc21290)

[4.3.2. 备库取消日志应用 29](#_Toc1729)

[4.3.3. 备库创建 standby redolog 30](#_Toc21861)

[4.3.4. 修改日志传输模式为 sync 30](#_Toc27123)

[4.3.5. 主库修改 log\_archive\_dest\_2 归档日志 30](#_Toc13516)

[4.3.6. 备库修改 log\_archive\_dest\_2 31](#_Toc23783)

[4.4. 最大性能切最大保护 32](#_Toc9231)

[4.4.1. 关库 32](#_Toc17373)

[4.4.2. 主库切最大保护 33](#_Toc10643)

[4.4.3. 主库插数据测试 33](#_Toc31290)

[4.4.4. 备库断开网络测试 34](#_Toc20316)

[主库插数据，commit 提交 hang 34](#_Toc21859)

[4.5. 最大保护切最高可用 34](#_Toc25341)

[5. 实时应用 Real-Time Apply 35](#_Toc2899)

[5.1. 启用 Real-Time Apply 35](#_Toc20050)

[5.2. 备库取消日志应用 35](#_Toc770)

[5.3. using current logfile 实时应用 35](#_Toc5847)

[5.4. 主库插入数据测试 35](#_Toc1238)

[6. switchover 计划内主备库切换 35](#_Toc16593)

[6.1. 查看主备库状态 36](#_Toc13228)

[6.2. 主库切物理备库 36](#_Toc31910)

[6.2.1. 重启原来主库到 mount 状态 36](#_Toc15177)

[6.2.2. 查看原来备库状态 36](#_Toc30745)

[6.2.3. 原备库切主库 37](#_Toc18224)

[6.2.4. open 打开新主库 37](#_Toc14918)

[6.2.5. 新备库应用日志 37](#_Toc11558)

[6.2.6. 取消日志应用 open 打开备库 38](#_Toc26765)

[6.3. 重新 switchover 主备切换 38](#_Toc20726)

[7. failover：主库意外宕机时，备库切换为主库 40](#_Toc977)

[7.1. 配置 Fast-start failover 40](#_Toc13102)

[8. 12c中DG新特性 40](#_Toc26807)

[9. 参考文档 41](#_Toc27375)

# Data Guard 容灾机制

## 概述

DG代表Data Guard，是一种用于实现数据库高可用性和灾难恢复的解决方案。Data Guard提供了一种数据保护和灾难恢复的机制，通过在主数据库和一个或多个备用数据库之间复制和同步数据，以确保在主数据库发生故障时可以快速切换到备用数据库并继续提供服务。

原理：主库将产生的 redo 日志传送到备库，备库对收到的 redo 日志进行应用。

### 主要功能包括：

* 数据保护：通过将主数据库的更改传输到备用数据库，确保数据的备份和保护。备用数据库可以位于远程位置，以提供更高的灾难恢复能力。
* 故障转移：当主数据库发生故障时，可以手动或自动地将备用数据库切换为主数据库，从而实现快速的故障转移和业务连续性。
* 可读取的备用数据库：备用数据库可以配置为可读取的，以便在主数据库不可用时，可以使用备用数据库提供只读查询功能。
* 数据库复制：Data Guard使用物理复制或逻辑复制将主数据库的更改传输到备用数据库。这确保备用数据库与主数据库保持同步，并可以在需要时快速切换。

## Standby Database（备库）的类型

### Physical Standby Database: 物理备库

Physical Standby Database是通过将主库的数据文件复制到备库来创建的。它是一个完全相同的副本，与主库保持同步。主库上的所有更改都会传输到备库并应用，以确保备库与主库的数据保持一致。物理备库可以提供高可用性和数据保护，以便在主库发生故障时快速切换到备库。

### Logical Standby Database: 逻辑备库

Logical Standby Database是通过将主库的日志传输到备库并在备库上重新执行来创建的。备库上的数据可以与主库上的数据不完全相同，因为备库上的操作可能会与主库上的操作有所不同。逻辑备库主要用于实时报表、查询和数据分析，因为它可以在备库上执行只读操作，而不会影响主库的性能。在主库发生故障时，逻辑备库可以转换为主库。

### Snapshot Standby Database: 快照备库

Snapshot Standby Database是物理备库的一种变体，它允许在备库上创建快照数据库，以供查询和报表使用，而不会中断主库的数据保护功能。快照备库可以定期将主库的数据应用到备库，但不会将备库的更改应用回主库。这使得快照备库可以提供实时查询和报表功能，同时仍然保持对主库的数据保护。

## 了解两种redo transport modes

### 同步（Synchronous）重做传输模式：

在同步传输模式下，主库在提交事务之前会等待至少一个物理备库确认已成功接收并应用了相应的重做数据。这确保了主库和备库之间的数据完全一致性，但也会导致主库的性能受到一定程度的影响，因为需要等待备库的确认。

### 异步（Asynchronous）重做传输模式：

在异步传输模式下，主库在提交事务后立即继续处理下一个事务，而不等待备库的确认。这样可以提高主库的性能，因为不需要等待备库的响应。然而，由于数据在传输到备库之前，主库和备库之间可能存在一定的延迟，因此备库的数据可能会稍微滞后于主库。这种模式适用于对数据一致性要求不高，但对性能要求较高的场景。

## 相关参数

### DB\_NAME：

指定数据库的全局唯一名称。它用于标识数据库实例和与之关联的其他数据库对象。

### DB\_UNIQUE\_NAME：

指定数据库的唯一名称，用于标识整个数据库系统。它通常用于在Data Guard配置中标识主库和备库。

### LOG\_ARCHIVE\_CONFIG：

指定归档日志的配置信息，包括归档日志的位置、归档日志的保留策略等。

### CONTROL\_FILES：

指定控制文件的位置和名称。控制文件是数据库的元数据文件，包含有关数据库结构和配置的信息。

### LOG\_ARCHIVE\_DEST\_n：

指定归档日志的目标位置。\_n是一个整数，可以有多个LOG\_ARCHIVE\_DEST参数，每个参数都对应一个归档日志目标。

### LOG\_ARCHIVE\_DEST\_STATE\_n：

指定LOG\_ARCHIVE\_DEST\_n参数的状态，可以是ENABLE（启用）或DISABLE（禁用）。

### REMOTE\_LOGIN\_PASSWORDFILE：

指定远程登录密码文件的类型。可以是NONE（不使用密码文件）、EXCLUSIVE（独占模式）或SHARED（共享模式）。

### LOG\_ARCHIVE\_FORMAT：

指定归档日志文件的命名格式。可以包含变量，如%t（线程号）、%s（日志序列号）等。

### LOG\_ARCHIVE\_MAX\_PROCESSES：

指定归档进程的最大数量。归档进程负责将日志传输到归档目标。

### FAL\_SERVER：

指定一个远程FAL（Fetch Archive Log）服务器的地址。FAL服务器用于在备库无法自动获取归档日志时，从主库获取归档日志。

### DB\_FILE\_NAME\_CONVERT：

指定主库和备库数据文件的名称转换规则。用于在备库上将主库的数据文件路径转换为备库上的路径。

### LOG\_FILE\_NAME\_CONVERT：

指定主库和备库日志文件的名称转换规则。用于在备库上将主库的日志文件路径转换为备库上的路径。

### STANDBY\_FILE\_MANAGEMENT：

指定备库上的文件管理方式。可以是AUTO（自动管理）或MANUAL（手动管理）。自动管理模式下，备库会自动创建和删除数据文件。手动管理模式下，需要手动管理备库上的文件。

## 什么是ADG

ADG是Oracle数据库中的Active Data Guard的缩写，它是一种高可用性和数据保护解决方案。ADG允许在主数据库和一个或多个备用数据库之间实时复制和同步数据，以提供高可用性和数据保护。与传统的备份和恢复解决方案不同，ADG可以实现零数据丢失和零停机时间的故障转移。

### ADG的主要特点包括：

* 实时数据复制：ADG使用物理数据保护技术，通过实时复制和同步数据来保持主数据库和备用数据库之间的一致性。
* 高可用性：在主数据库发生故障或不可用时，ADG可以自动将备用数据库切换为可读写模式，以提供连续的数据库服务。
* 数据保护：ADG提供了数据保护的解决方案，可以防止数据丢失和损坏。备用数据库可以用于恢复数据，以保护数据库免受硬件故障、自然灾害或人为错误的影响。
* 查询分担：ADG允许在备用数据库上执行只读查询操作，以减轻主数据库的负载，提高性能。
* 自动故障转移：当主数据库不可用时，ADG可以自动将备用数据库切换为主数据库，以实现快速的故障转移和恢复。

## 什么是MAA

MAA是Oracle数据库的最佳实践架构（Maximum Availability Architecture）的缩写。MAA是一种综合性的架构设计，旨在提供最高级别的可用性、可靠性和性能，以满足关键业务系统的需求。

### MAA的主要组件包括：

* 数据库备份和恢复：MAA建议使用Oracle的备份和恢复解决方案，如RMAN（Recovery Manager）来实现数据库的备份和恢复。RMAN提供了高度可靠和高效的备份和恢复功能。
* 数据库复制和同步：MAA建议使用Oracle Data Guard来实现数据库的复制和同步。Data Guard提供了实时的数据复制和故障转移功能，可以确保数据库的连续可用性和数据保护。
* 高可用性集群：MAA建议使用Oracle Real Application Clusters（RAC）来实现数据库的高可用性。RAC是一种集群技术，可以将多个数据库服务器组合成一个逻辑数据库，提供高可用性和可伸缩性。
* 数据库性能优化：MAA建议使用Oracle的性能优化工具和技术，如AWR（Automatic Workload Repository）和ADDM（Automatic Database Diagnostic Monitor），来监控和优化数据库的性能。

## 日志

### 日志存储

redo log buffer 日志缓冲区

onlind redo logfiles 联机日志文件

archived redo logfiles 归档日志文件

### 日志传送

async （noaffirm）异步：主库将日志传送到备库，备库不需要返回确认。

sync (affirm) 同步：主库将日志传送到备库，备库将日志写入 standby redo

logfiles,返回确认给主库。主库收到确认后，完成事务的提交。

传送进程 LNS:Log Network Services

### 日志接收

备库接收主库传送的日志

如果传送方式是 sync,需将日志写入 standby redo logfiles 并返回确认。

接收进程：RFS:Remote file services

## 日志应用

### Redo Apply-物理 DG

备库对接收到的日志进行块级别日志应用；

备库和主库具有相同的 dbid,dbname，不同的 db\_unique\_name

备库只能 open read only

MRP 进程进行 redo apply：managed standby recovery process

### SQL Apply-逻辑 DG

备库将从主库接收到日志转换成 SQL 语句进行执行

备库和主库是独立的库，dbname,dbid 不同

LSP 进程进行 sql apply：logical standby process

## 如何监控DG运行状态

### 相关视图及命令

视图

* v$dataguard\_stats视图：提供有关DG运行状态的统计信息，如传输延迟、应用延迟、重做传输速率等。
* v$dataguard\_status视图：提供有关DG配置和运行状态的详细信息，如主库和备库的角色、传输模式、应用延迟等。

命令

* show configuration命令：在SQL\*Plus中执行该命令可以查看DG的当前配置信息，包括主库和备库的角色、传输模式等。
* show database命令：在RMAN（Recovery Manager）中执行该命令可以查看DG的当前状态，包括主库和备库的角色、传输模式、应用延迟等。
* DGMGRL（Data Guard Broker Command-Line Interface）：DGMGRL是一个命令行工具，可以用于管理和监控DG配置。通过DGMGRL，可以执行各种命令来查看DG的状态、切换角色、启用/禁用重做应用等。

### 主库环境检查

主库DG环境的保护模式检查

select database\_role,protection\_mode,protection\_level from v$database;

## GAP

### 什么是GAP

GAP（Guaranteed Availability Protection）是Oracle Data Guard中的一个概念，用于描述主库和备库之间的数据同步延迟。当主库发生故障或不可用时，备库会接管并成为新的主库，但在此过程中可能会存在数据同步的延迟。

### 如何处理GAP

* 使用实时应用（Real-Time Apply）：在Oracle Data Guard中，可以启用实时应用功能，使备库能够实时应用主库的日志，从而减少数据同步延迟。这样可以尽可能地减小GAP。
* 使用远程同步（Far Sync）：远程同步是Oracle Data Guard 12c引入的一个功能，可以在不同的数据中心之间实现实时的同步，以提供更高的灾难恢复能力。通过使用远程同步，可以减小主库和备库之间的数据同步延迟，从而减小GAP。
* 使用快速故障转移（Fast Failover）：快速故障转移是Oracle Data Guard 12c引入的一个功能，可以在主库宕机时更快地将备库切换为新的主库。通过使用快速故障转移，可以减小主库故障时的业务中断时间，从而减小GAP。
* 定期监控和调整配置：定期监控主库和备库之间的数据同步延迟，并根据实际情况调整配置，以减小GAP。例如，可以增加网络带宽、优化数据库参数配置等。

## 常见报错处理

### ORA-10458: standby database requires recovery。ORA-01196: ORA-01110:

昨晚电脑自动关机，第二天启动standby数据库时报错

ORA-10458: standby database requires recovery

ORA-01196: file 1 is inconsistent due to a failed media recovery session

ORA-01110: data file 1: '/u01/app/oracle/oradata/prod1/system01.dbf'

原因强制关机导致日志记录的SCN与控制文件不符，不能完成checkpoint事件，需要进行介质恢复。

重启后先开监听！！！！！！

主备应用有延迟：

select thread#, max(sequence#) from v$archived\_log group by thread#;

DG比主库的 SEQUENCE慢，主从同步有问题。

* 处理步骤：

startup force mount

进行日志恢复

alter database recover managed standby database disconnect from session;

取消日志恢复

alter database recover managed standby database cancel;

打开数据库，并开启实时应用

alter database open;

alter database recover managed standby database using current logfile disconnect from session;

报错ORA-01153。处理

SYS@prod1> alter database recover managed standby database using current logfile disconnect from session;

alter database recover managed standby database using current logfile disconnect from session

\*

ERROR at line 1:

ORA-01153: an incompatible media recovery is active

SYS@prod1> alter database recover managed standby database cancel;

Database altered.

SYS@prod1> alter database recover managed standby database using current logfile disconnect from session;

Database altered.

查看是否实时应用，查看已经应用

set line 199

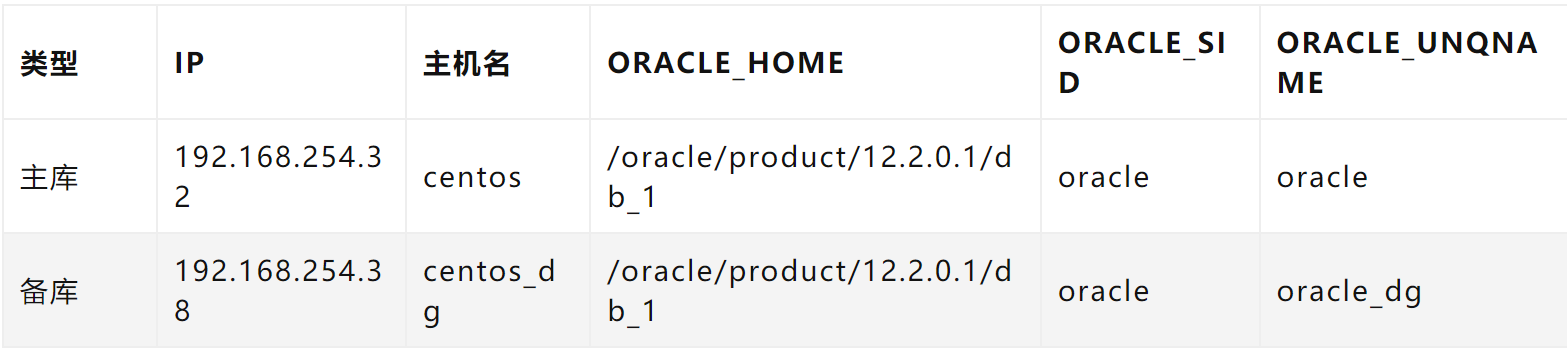
col dest\_name for a50

select DEST\_NAME , STATUS , RECOVERY\_MODE from v$archive\_dest\_status;

# 创建物理 DG

## 需求规划

|  |  |  |
| --- | --- | --- |
|  | 主库 | 备库 |
| db\_name | prod1 | prod1 |
| db\_unique\_name | prod1 | test1 |
| net service name | prod1 | test1 |
| SID/instance\_name | prod1 | test1 |
| service\_name | prod1.enmoedu.com | test1.enmoedu.com |



备库文件目录

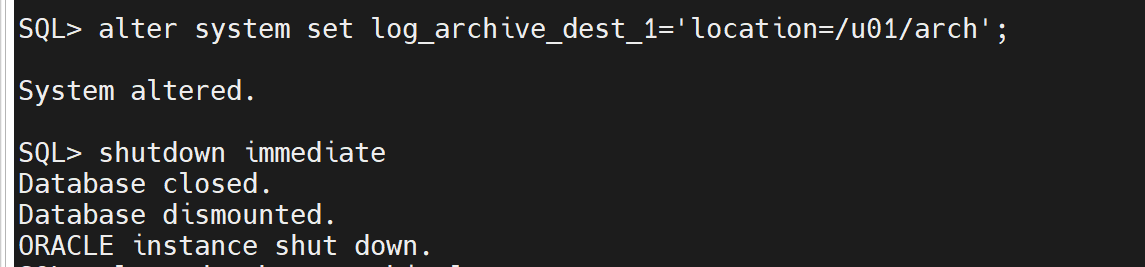
/u01/app/oracle/oradata/test1

## 主库开归档

[oracle@prod1 ~]$ mkdir /u01/arch

SYS@prod1>

alter system set log\_archive\_dest\_1='location=/u01/arch';

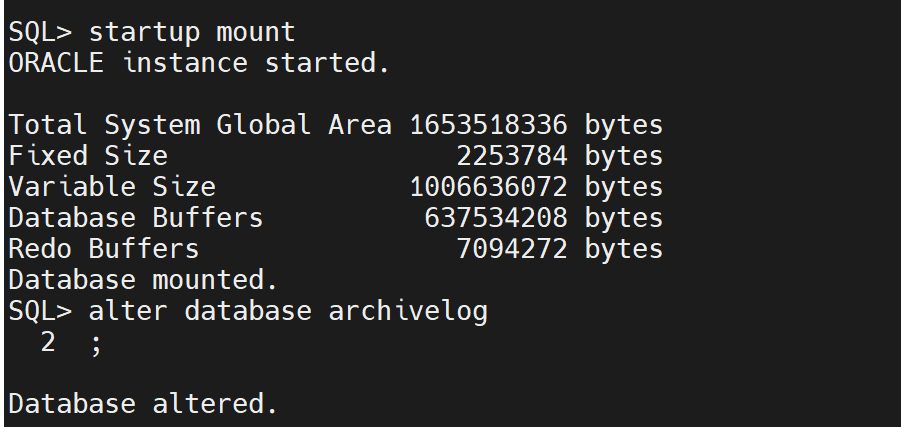


shutdown immediate;

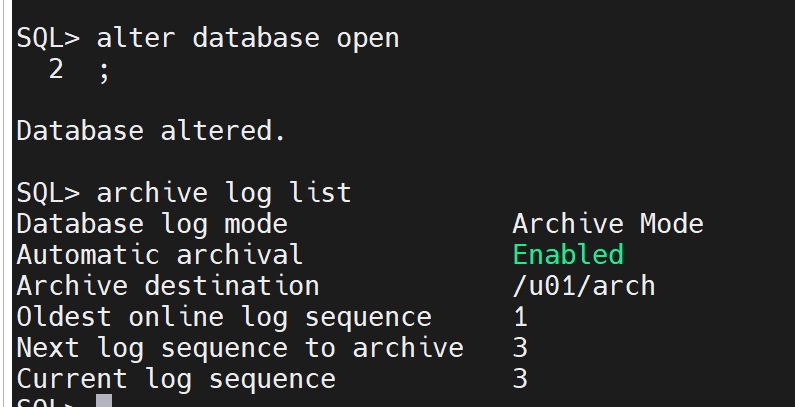
startup mount

alter database archivelog;

alter database open;



SYS@prod1>archive log list;



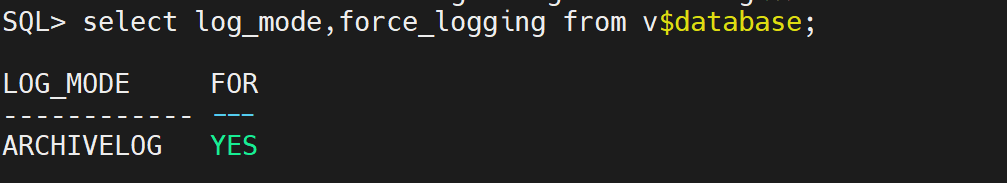
## FORCE LOGGING

当数据库设置为FORCE LOGGING时，所有的数据修改操作都会被强制写入日志文件，无论是否启用了日志记录模式。

启用FORCE LOGGING的主要目的是确保数据库的完整性和一致性。通过强制所有数据修改操作都写入日志文件，可以保证在数据库发生故障或崩溃时，可以通过日志重做来进行恢复操作，从而避免数据丢失和不一致性。

SYS@prod1>ALTER DATABASE FORCE LOGGING;

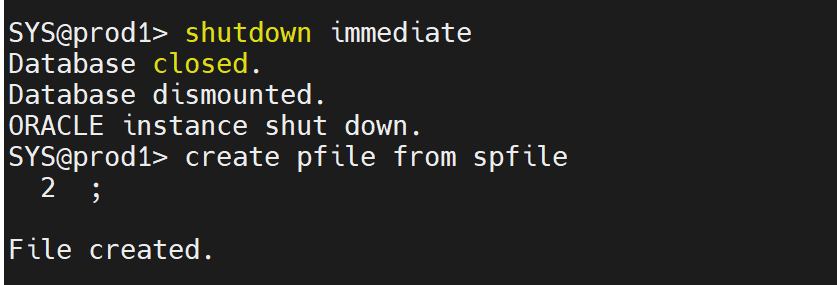
SYS@prod1>select log\_mode,force\_logging from v$database;



## 创建主库 pfile 文件

SYS@prod1>shutdown immediate;

SYS@prod1>create pfile from spfile;



### 修改 pfile 文件

[oracle@prod1 ~]$ cd $ORACLE\_HOME/dbs

[oracle@prod1 dbs]$ vi initprod1.ora

DB\_UNIQUE\_NAME=prod1

LOG\_ARCHIVE\_CONFIG='DG\_CONFIG=(prod1,test1)'

LOG\_ARCHIVE\_DEST\_1=

'LOCATION=/u01/arch

VALID\_FOR=(ALL\_LOGFILES,ALL\_ROLES)

DB\_UNIQUE\_NAME=prod1'

LOG\_ARCHIVE\_DEST\_2=

'SERVICE=test1 ASYNC

VALID\_FOR=(ONLINE\_LOGFILES,PRIMARY\_ROLE)

DB\_UNIQUE\_NAME=test1'

LOG\_ARCHIVE\_DEST\_STATE\_1=ENABLE

LOG\_ARCHIVE\_DEST\_STATE\_2=ENABLE

LOG\_ARCHIVE\_FORMAT=%t\_%s\_%r.arc

LOG\_ARCHIVE\_MAX\_PROCESSES=3

FAL\_SERVER=test1

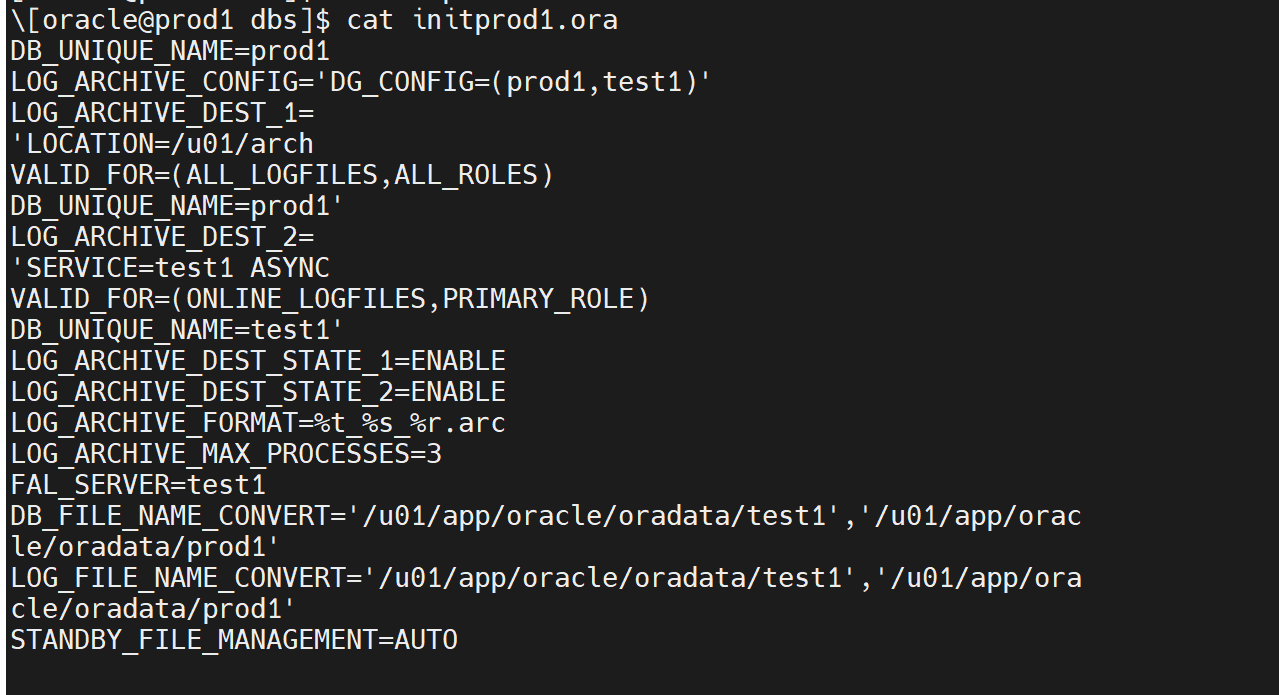
DB\_FILE\_NAME\_CONVERT='/u01/app/oracle/oradata/test1','/u01/app/orac

le/oradata/prod1'

LOG\_FILE\_NAME\_CONVERT='/u01/app/oracle/oradata/test1','/u01/app/ora

cle/oradata/prod1'

STANDBY\_FILE\_MANAGEMENT=AUTO

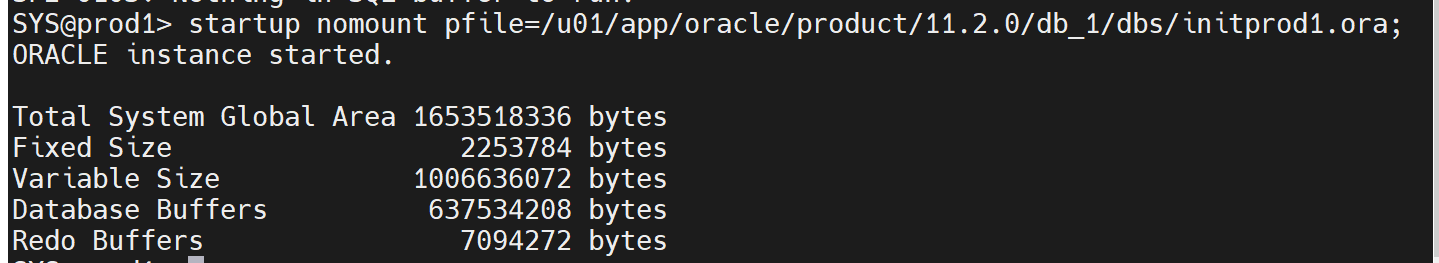


### pfile 起库测试

SYS@prod1>

startup nomount pfile=$ORACLE\_HOME/dbs/initprod1.ora;

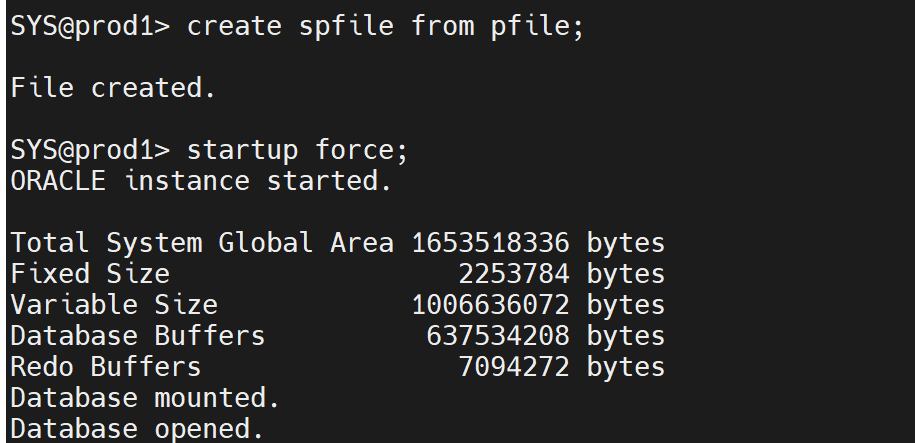
show parameter name;



### 创建 spfile,重新启库

SYS@prod1>create spfile from pfile;

SYS@prod1>startup force;



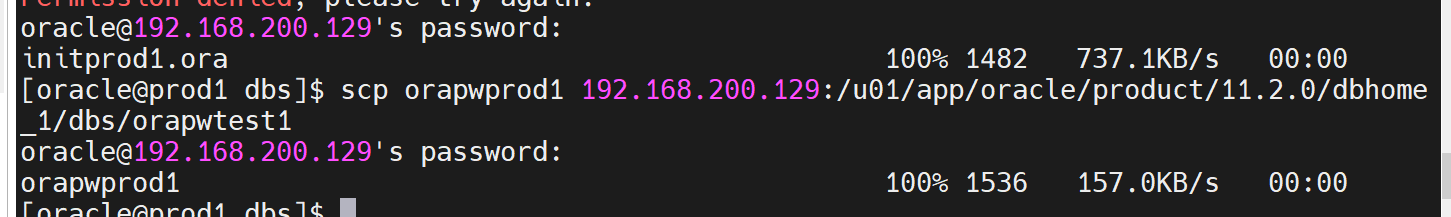
## 备库参数文件、密码文件

### 拷贝参数文件、密码文件到备库

[oracle@prod1 dbs]$

scp initprod1.ora 192.168.200.129:/u01/app/oracle/product/11.2.0/dbhome\_1/dbs/inittest1.ora

scp orapwprod1 192.168.200.129:/u01/app/oracle/product/11.2.0/dbhome\_1/dbs/orapwtest1



### 修改备库参数文件

[oracle@ocm2 dbs]$ vi inittest1.ora

删除 PROD.开头的

:%s/test1/AAAA/g

:%s/prod1/test1/g

:%s/AAAA/prod1/g

修改\*.db\_name='prod1'

[oracle@ocm2 dbs]$ vi inittest1.ora

DB\_UNIQUE\_NAME=test1

LOG\_ARCHIVE\_CONFIG='DG\_CONFIG=(test1,prod1)'

LOG\_ARCHIVE\_DEST\_1=

'LOCATION=/u01/arch

VALID\_FOR=(ALL\_LOGFILES,ALL\_ROLES)

DB\_UNIQUE\_NAME=test1'

LOG\_ARCHIVE\_DEST\_2=

'SERVICE=prod1 ASYNC

VALID\_FOR=(ONLINE\_LOGFILES,PRIMARY\_ROLE)

DB\_UNIQUE\_NAME=prod1'

LOG\_ARCHIVE\_DEST\_STATE\_1=ENABLE

LOG\_ARCHIVE\_DEST\_STATE\_2=ENABLE

LOG\_ARCHIVE\_FORMAT=%t\_%s\_%r.arc

LOG\_ARCHIVE\_MAX\_PROCESSES=3

FAL\_SERVER=prod1

DB\_FILE\_NAME\_CONVERT='/u01/app/oracle/oradata/prod1','/u01/app/orac

le/oradata/test1'

LOG\_FILE\_NAME\_CONVERT='/u01/app/oracle/oradata/prod1','/u01/app/ora

cle/oradata/test1'

STANDBY\_FILE\_MANAGEMENT=AUTO

test1.\_\_db\_cache\_size=687865856

test1.\_\_java\_pool\_size=16777216

test1.\_\_large\_pool\_size=33554432

test1.\_\_oracle\_base='/u01/app/oracle'#ORACLE\_BASE set from environment

test1.\_\_pga\_aggregate\_target=671088640

test1.\_\_sga\_target=989855744

test1.\_\_shared\_io\_pool\_size=0

test1.\_\_shared\_pool\_size=234881024

test1.\_\_streams\_pool\_size=0

\*.audit\_trail='db'

\*.audit\_file\_dest='/u01/app/oracle/admin/test1/adump'

\*.compatible='11.2.0.4.0'

\*.control\_files='/u01/app/oracle/oradata/test1/control01.ctl','/u01/app/oracle/oradata/test1/control02.ctl'

\*.db\_block\_size=8192

\*.db\_domain=''

\*.db\_name='prod1'

\*.diagnostic\_dest='/u01/app/oracle'

\*.dispatchers='(PROTOCOL=TCP) (SERVICE=test1XDB)'

\*.log\_archive\_dest\_1='location=/u01/arch'

\*.memory\_target=1647312896

\*.open\_cursors=300

\*.processes=150

\*.remote\_login\_passwordfile='EXCLUSIVE'

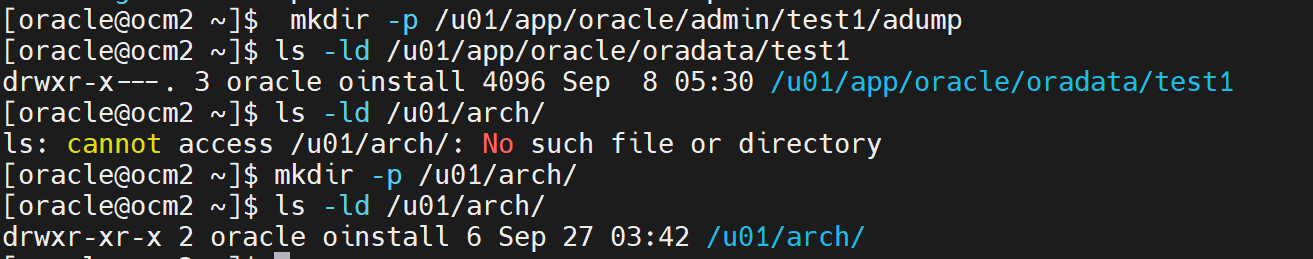
\*.undo\_tablespace='UNDOTBS1'

### 备库创建目录

[oracle@ocm2 ~]$ mkdir -p /u01/app/oracle/admin/test1/adump

[oracle@ocm2 ~]$ ls -ld /u01/app/oracle/oradata/test1

[oracle@ocm2 ~]$ ls -ld /u01/arch/



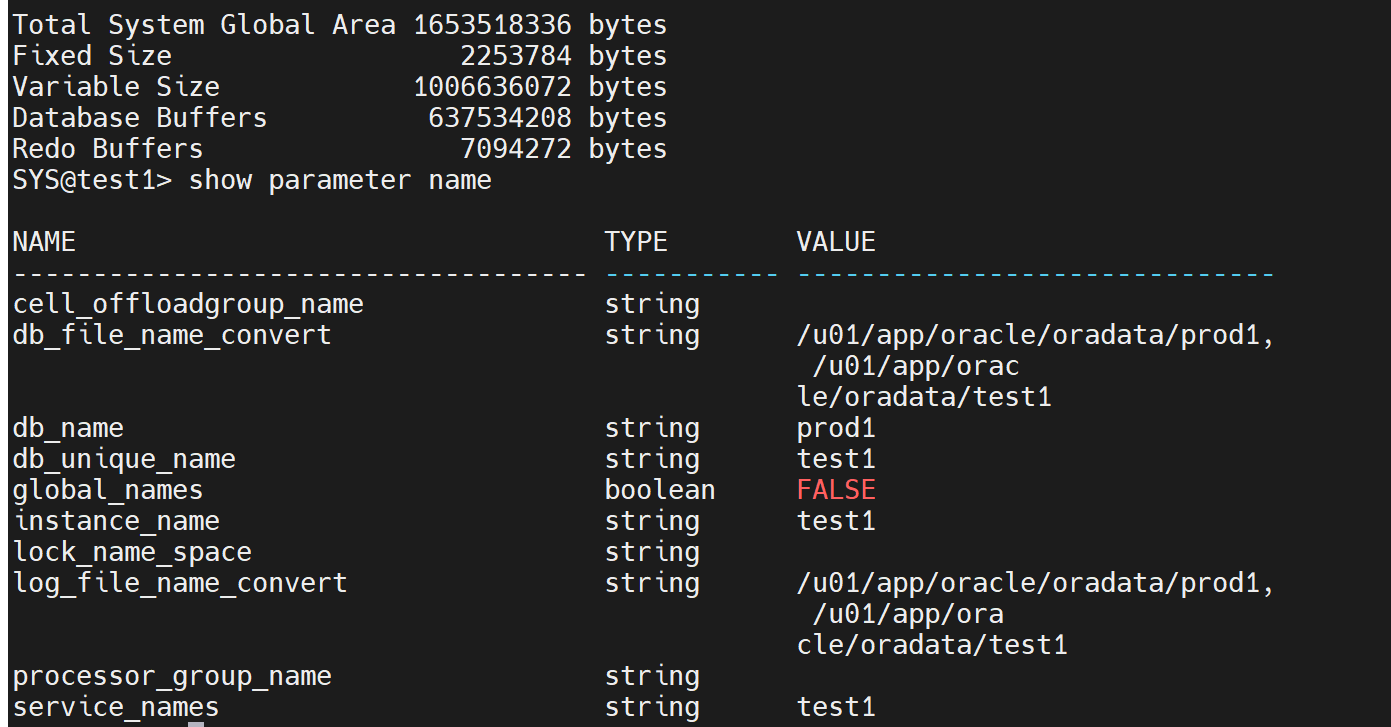
### 备库启动实例到 nomount 查看参数确认

[oracle@ocm2 ~]$ export ORACLE\_SID=test1

[oracle@ocm2 ~]$ sqlplus / as sysdba

SYS@test1>startup nomount pfile=$ORACLE\_HOME/dbs/inittest1.ora;

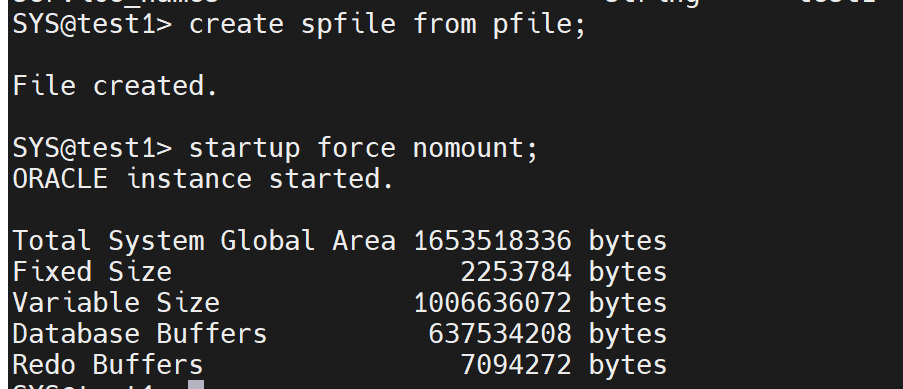
SYS@test1>show parameter name



### 备库创建 spfile 重启实例到 nomount

SYS@test1>create spfile from pfile;

SYS@test1>startup force nomount;



## 配置主、备库网络

### 主库静态监听

[oracle@prod1 ~]$ cd $ORACLE\_HOME/network/admin

[oracle@prod1 admin]$ vi listener.ora

[oracle@prod1 admin]$ cat listener.ora

LISTENER =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = prod1.example.com)(PORT = 1521))

)

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(SID\_NAME = prod1)

(ORACLE\_HOME = /u01/app/oracle/product/11.2.0/db\_1)

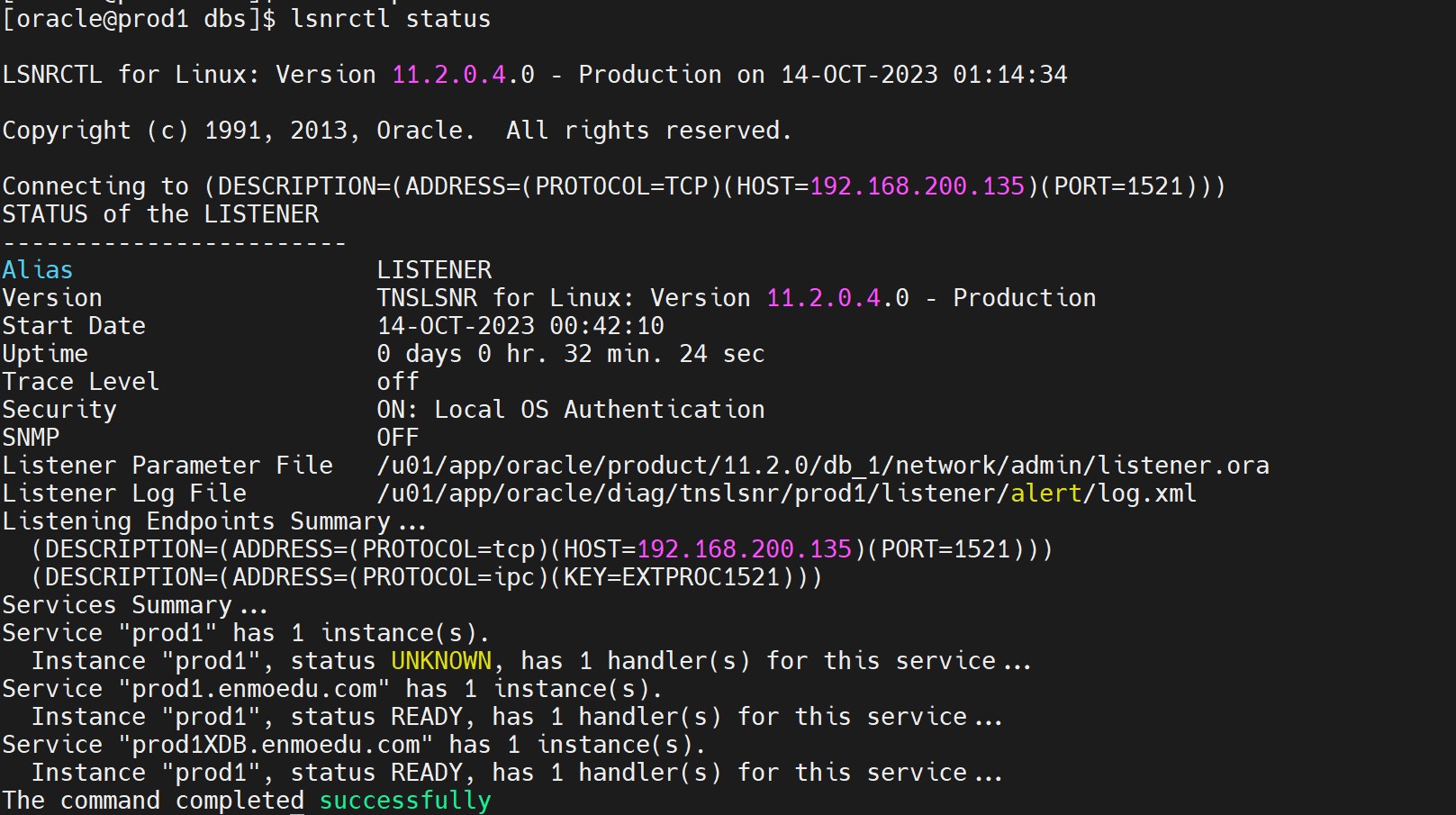
(GLOBAL\_DBNAME=prod1.enmoedu.com)

)

)

[oracle@prod1 admin]$ lsnrctl reload

[oracle@prod1 admin]$ lsnrctl status



## 备库静态 listener

cd $ORACLE\_HOME/network/admin

$ vi listener.ora

LISTENER2 =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = ocm2)(PORT = 1521))

)

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(SID\_NAME = test1)

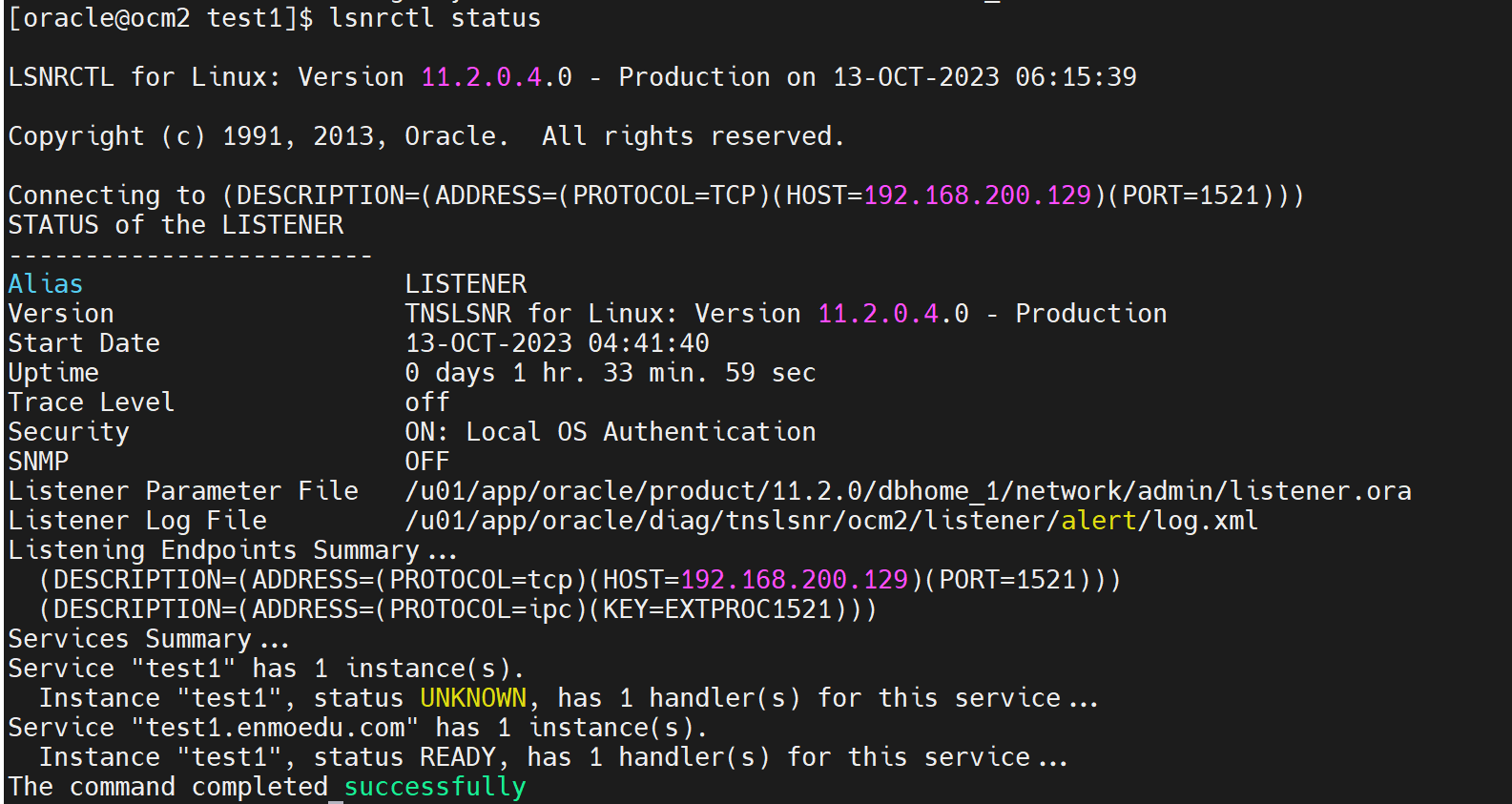
(ORACLE\_HOME = /u01/app/oracle/product/11.2.0/dbhome\_1)

)

)

[oracle@ocm2 admin]$ lsnrctl stop

[oracle@ocm2 admin]$ lsnrctl start



### 主备库 tnsnames.ora

[oracle@prod1 admin]$ cp tnsnames.ora tnsnames.ora.bak

[oracle@prod1 admin]$ vi tnsnames.ora

主库 tnsnames.ora

prod1 =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST =192.168.200.135)(PORT = 1521))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = prod1.enmoedu.com)

)

)

test1 =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST =192.168.200.129)(PORT = 1521))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = test1.enmoedu.com)

)

)

备库 tnsnames.ora

### 测试网络

[oracle@prod1 admin]$ sqlplus sys/oracle@prod1 as sysdba

[oracle@prod1 admin]$ sqlplus sys/oracle@test1 as sysdba

[oracle@ocm2 admin]$ sqlplus sys/oracle@prod1 as sysdba

[oracle@ocm2 admin]$ sqlplus sys/oracle@test1 as sysdba

## rman duplicate 创建物理备库

### 主备库打开告警日志

[oracle@prod1 ~]$ tail -f

/u01/app/oracle/diag/rdbms/prod1/prod1/trace/alert\_prod1.log

[oracle@ocm2 ~]$ tail -f

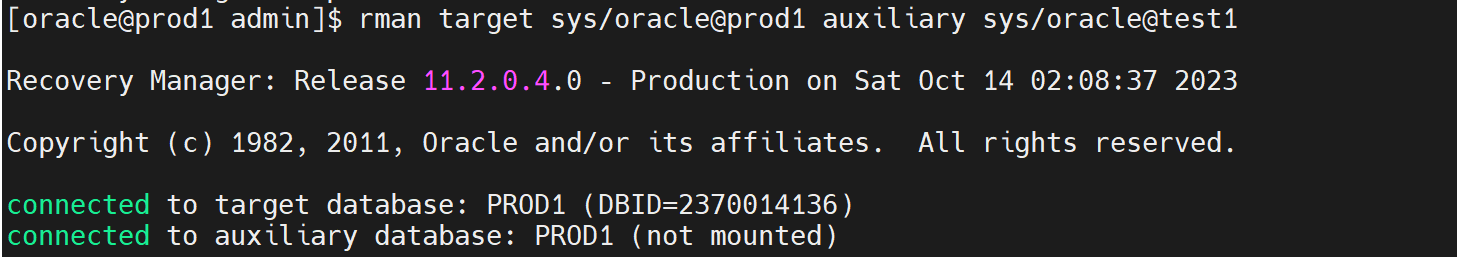
/u01/app/oracle/diag/rdbms/test1/test1/trace/alert\_test1.log

### rman 登录主备库

主库open

备库nomount

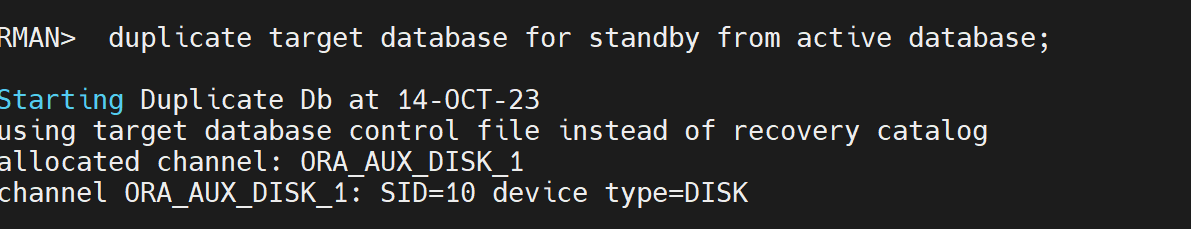
[oracle@prod1 ~]$ rman target sys/oracle@prod1 auxiliary sys/oracle@test1

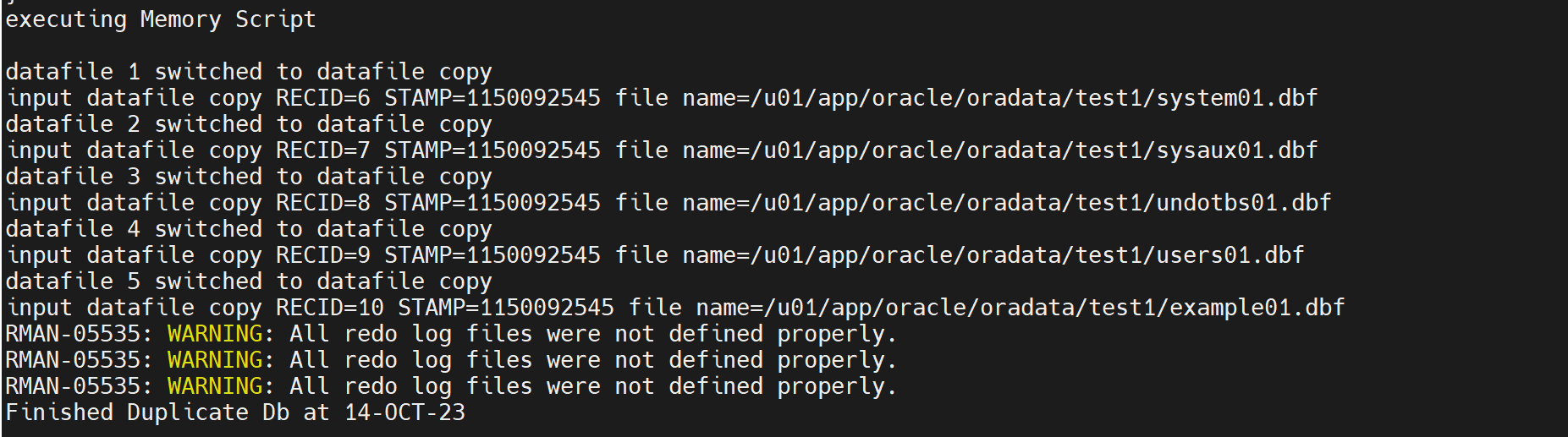


### 执行 rman duplicate

从活动数据库创建一个备用数据库。

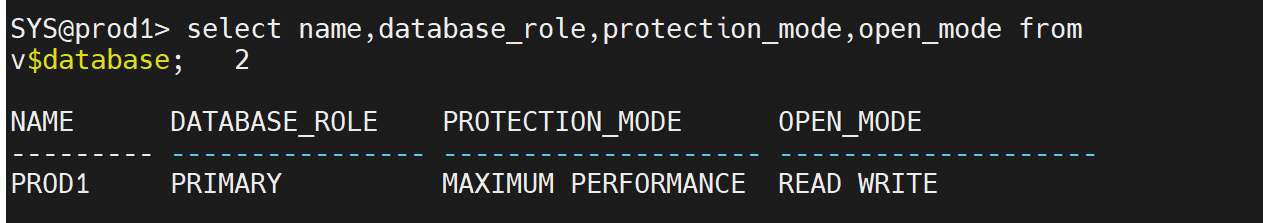
RMAN> duplicate target database for standby from active database;



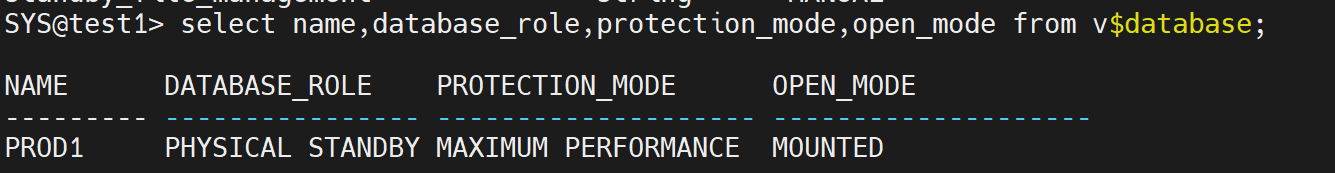


### 查看主备库状态

SYS@prod1>select name,database\_role,protection\_mode,open\_mode from v$database;



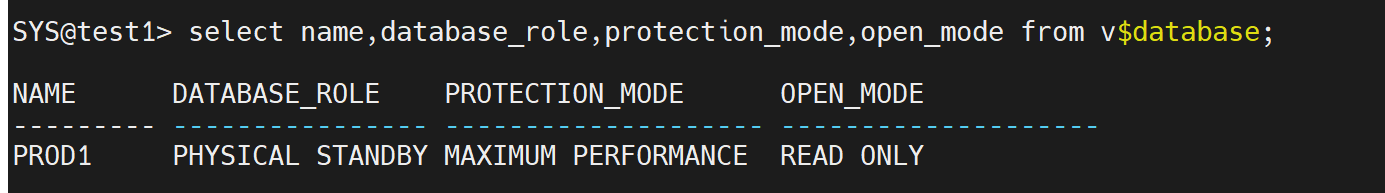
SYS@test1>select name,database\_role,protection\_mode,open\_mode from v$database;



### open 打开备库

SYS@test1>alter database open;

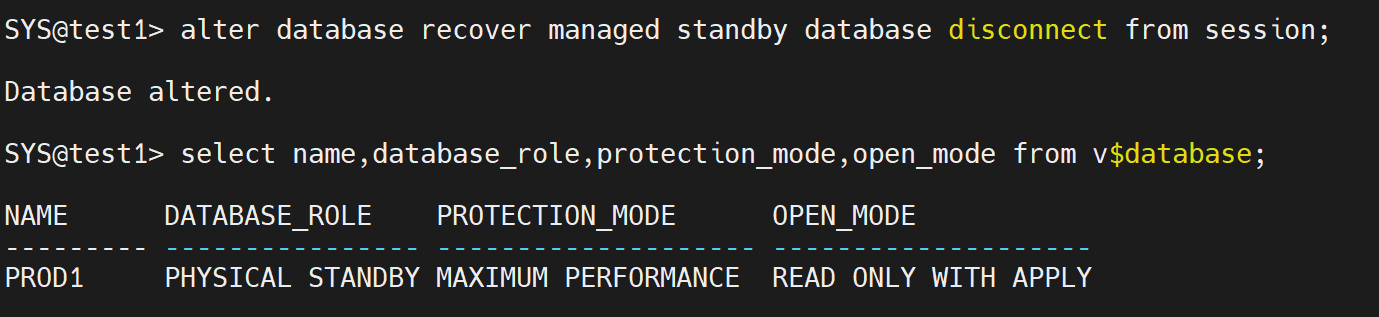
SYS@test1>select name,database\_role,protection\_mode,open\_mode from v$database;



## 备库启动 MRP 进程 应用日志

SYS@test1>alter database recover managed standby database disconnect from session;

SYS@test1>select name,database\_role,protection\_mode,open\_mode from v$database;



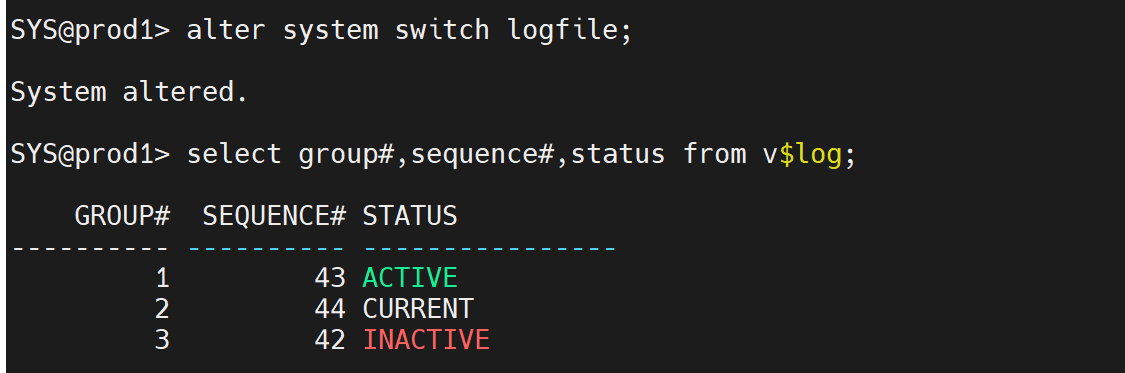
告警日志信息

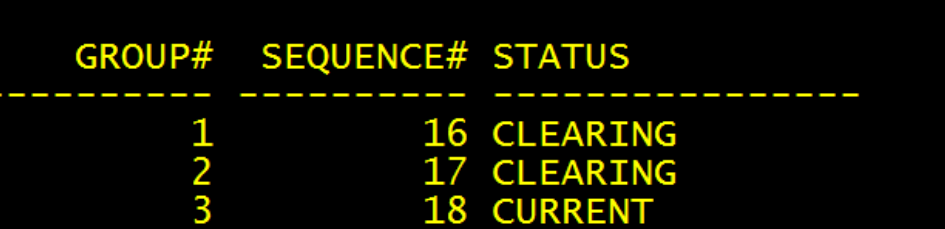
## 测试

### 主库切归档 测试

SYS@prod1>alter system switch logfile;

SYS@prod1>select group#,sequence#,status from v$log;





### scott 建表插数据测试

conn bob/bob

BOB@prod1>

CREATE TABLE EMP

(EMPNO NUMBER(4) CONSTRAINT PK\_EMP PRIMARY KEY,

ENAME VARCHAR2(10),

JOB VARCHAR2(9),

MGR NUMBER(4),

HIREDATE DATE,

SAL NUMBER(7,2),

COMM NUMBER(7,2),

DEPTNO NUMBER(2) );

INSERT INTO EMP VALUES

(7369,'SMITH','CLERK',7902,to\_date('17-12-1980','dd-mm-yyyy'),800,NULL,20);

INSERT INTO EMP VALUES

(7499,'ALLEN','SALESMAN',7698,to\_date('20-2-1981','dd-mm-yyyy'),1600,300,30);

INSERT INTO EMP VALUES

(7521,'WARD','SALESMAN',7698,to\_date('22-2-1981','dd-mm-yyyy'),1250,500,30);

INSERT INTO EMP VALUES

(7566,'JONES','MANAGER',7839,to\_date('2-4-1981','dd-mm-yyyy'),2975,NULL,20);

INSERT INTO EMP VALUES

(7654,'MARTIN','SALESMAN',7698,to\_date('28-9-1981','dd-mm-yyyy'),1250,1400,30);

INSERT INTO EMP VALUES

(7698,'BLAKE','MANAGER',7839,to\_date('1-5-1981','dd-mm-yyyy'),2850,NULL,30);

INSERT INTO EMP VALUES

(7782,'CLARK','MANAGER',7839,to\_date('9-6-1981','dd-mm-yyyy'),2450,NULL,10);

INSERT INTO EMP VALUES

(7788,'SCOTT','ANALYST',7566,to\_date('12-06-1987','dd-mm-yyyy')-85,3000,NULL,20);

INSERT INTO EMP VALUES

(7839,'KING','PRESIDENT',NULL,to\_date('17-11-1981','dd-mm-yyyy'),5000,NULL,10);

INSERT INTO EMP VALUES

(7844,'TURNER','SALESMAN',7698,to\_date('8-9-1981','dd-mm-yyyy'),1500,0,30);

INSERT INTO EMP VALUES

(7876,'ADAMS','CLERK',7788,to\_date('13-06-1987','dd-mm-yyyy')-51,1100,NULL,20);

INSERT INTO EMP VALUES

(7900,'JAMES','CLERK',7698,to\_date('3-12-1981','dd-mm-yyyy'),950,NULL,30);

INSERT INTO EMP VALUES

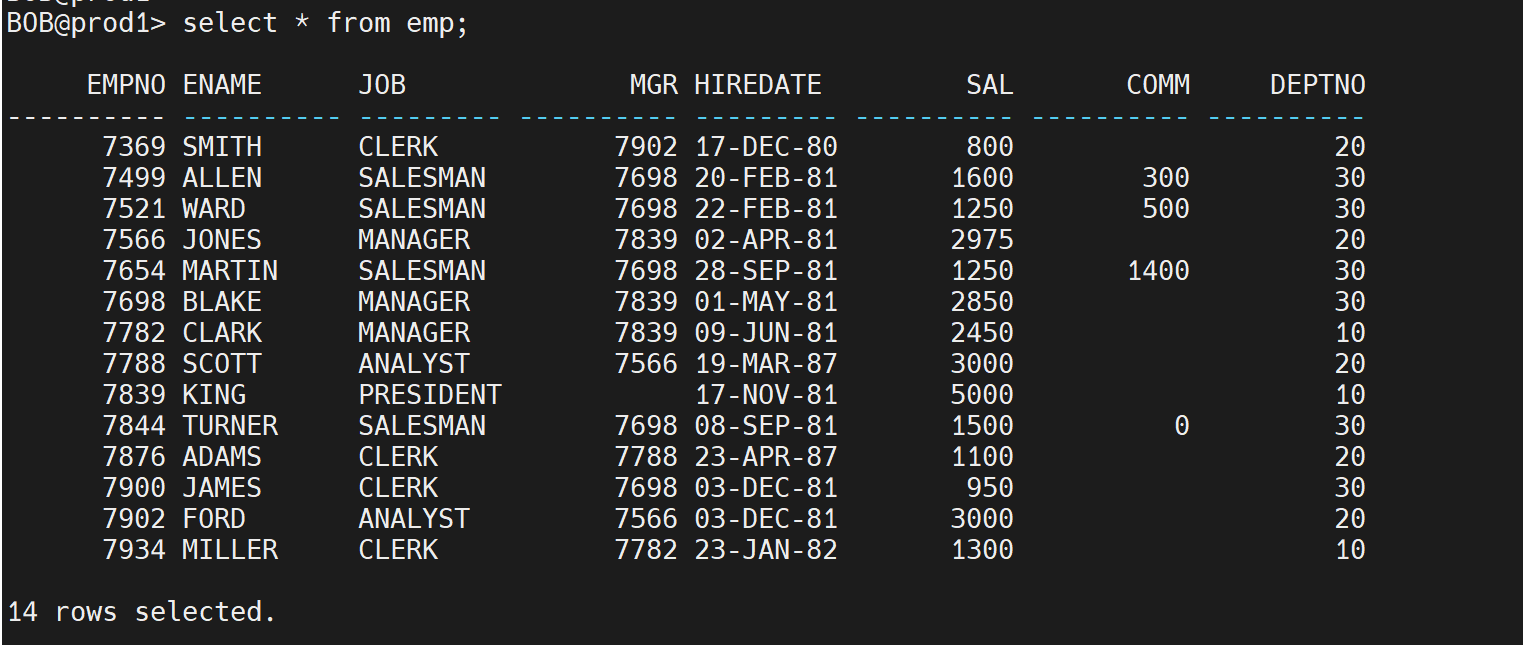
(7902,'FORD','ANALYST',7566,to\_date('3-12-1981','dd-mm-yyyy'),3000,NULL,20);

INSERT INTO EMP VALUES

(7934,'MILLER','CLERK',7782,to\_date('23-1-1982','dd-mm-yyyy'),1300,NULL,10);

commit;

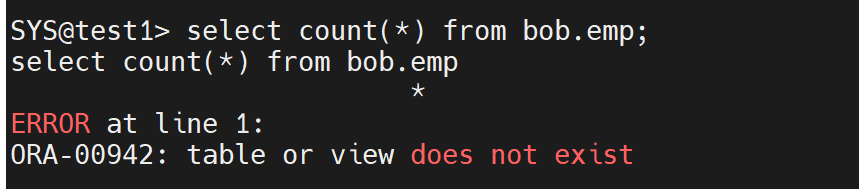
select count(\*) from emp;



备库查看

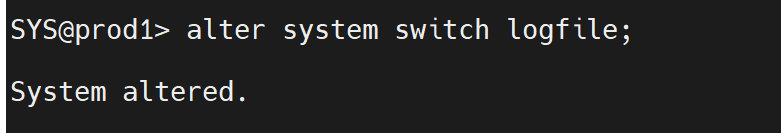
SYS@test1>select count(\*) from bob.emp;

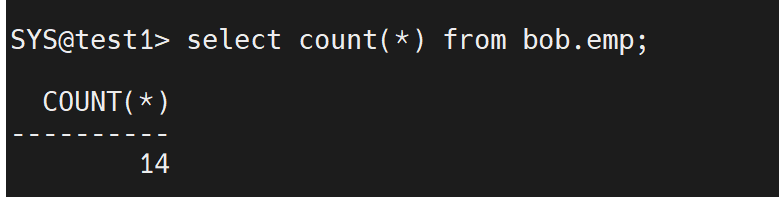
报错



当前模式下，没有启用 Real time apply,默认只对归档日志进行应用。

主库切日志组，备库再次查看。





## 报错

### 报错1

RMAN-00571: ===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS ===============

RMAN-00571: ===========================================================

RMAN-03002: failure of Duplicate Db command at 10/14/2023 01:14:08

RMAN-05501: aborting duplication of target database

RMAN-03015: error occurred in stored script Memory Script

RMAN-03009: failure of backup command on ORA\_DISK\_1 channel at 10/14/2023 01:14:08

ORA-17628: Oracle error 19505 returned by remote Oracle server

注意DB\_FILE\_NAME\_CONVERT参数，要么是没有设置，或者设置有问题，

我的报错是这个参数设置时有一个换行符

### 报错2

RMAN-00571: ===========================================================

RMAN-00569: =============== ERROR MESSAGE STACK FOLLOWS ===============

RMAN-00571: ===========================================================

RMAN-03002: failure of Duplicate Db command at 10/14/2023 01:03:34

RMAN-05501: aborting duplication of target database

RMAN-03015: error occurred in stored script Memory Script

RMAN-03009: failure of backup command on ORA\_DISK\_1 channel at 10/14/2023 01:03:34

ORA-17629: Cannot connect to the remote database server

ORA-17627: ORA-01017: invalid username/password; logon denied

ORA-17629: Cannot connect to the remote database server

rman登陆的时候有问题

之前用rman target / auxiliary sys/oracle@sbdb1

修改后使用rman target sys/oracle@prod1 auxiliary sys/oracle@test1

### 报错3

RMAN-05535: WARNING: All redo log files were not defined properly.

ORACLE error from auxiliary database: ORA-01511: error in renaming log/data files

ORA-01275: Operation RENAME is not allowed if standby file management is automatic.

命令执行报错，提示说standbyfile maangement参数为自动，自动情况下不允许修改，好吧，听它的，修改成手动的，这样我们就可以移动它的目录地址了

show parameter standby;

alter system set standby\_file\_management = MANUAL;

startup force nomount

# 快照DG

在物理 DG 基础上构建可读写的测试环境；

物理 DG 转换为快照 DG 时生成 snapshot,创建 restore point,

可以不开启 flashback database,但需要设置 Fast recovery area;

快照 DG 可以进行读写，能够接收主库日志但不应用日志；

快照 DG 切回物理 DG 时，通过 flashback database 回到还原点。

场景：生产上，有一个测试任务，使用快照备库先测试，测试完，测试成功再去主库上测试

。通过 flashback database 回到还原点

## 设置 Fast recovery area

可以不启动 flashback databaes,但需要设置 Fast recovery area

[oracle@ocm2 ~]$ mkdir /u01/flash

SYS@test1>show parameter db\_recovery

SYS@test1>alter system set db\_recovery\_file\_dest\_size=4G;

SYS@test1>alter system set db\_recovery\_file\_dest='/u01/flash';

SYS@test1>show parameter db\_recovery

## 备库取消日志应用

SYS@test1>recover managed standby database cancel;

## 备库关库启动到 mount 状态

SYS@test1>shutdown immediate;

SYS@test1>startup mount;

SYS@test1>select name,database\_role,protection\_mode,open\_mode from

v$database;

## 物理备库切 snapshot standby

SYS@test1>alter database convert to snapshot standby;

select name,database\_role,protection\_mode,open\_mode from v$database;

alter database open;

select name,database\_role,protection\_mode,open\_mode from v$database;

## 快照备库写测试

SYS@test1>create table scott.emp2 as select \* from scott.emp;

SYS@test1>select count(\*) from scott.emp2;

## 主库插数据切日志测试

主库 scott.emp1 查数据，提交，切日志组

SYS@prod1>insert into scott.emp1 select \* from scott.emp1;

SYS@prod1>commit;

SYS@prod1>select count(\*) from scott.emp1;

SYS@prod1>alter system switch logfile;

* 备库查看

备库接收日志，但不应用日志。

select count(\*) from scott.emp1;

## 快照 DG 切回物理 DG

SYS@test1>

select name,database\_role,protection\_mode,open\_mode from v$database;

关闭备库，启动到 mount 状态

SYS@test1>

shutdown immediate;

startup mount;

切换到物理备库

SYS@test1>alter database convert to physical standby;

SYS@test1>select status from v$instance;

## 关闭备库重新启动

SYS@test1>shutdown immediate;

SYS@test1>startup

SYS@test1>select name,database\_role,protection\_mode,open\_mode from

v$database;

## 备库应用日志，查看

SYS@test1>

recover managed standby database disconnect;

select name,database\_role,protection\_mode,open\_mode from v$database;

查看表中数据，快照 DG 期间备库建的表已不存在，备库已经应用日志与主库同步

SYS@test1>select count(\*) from scott.emp2;

SYS@test1>select count(\*) from scott.emp1;

告警日志信息

# 三种保护模式

## 保护模式

### 最大保护 maximum protection

最大保护模式

-1) 这种模式提供了最高级别的数据保护能力;- 2) 要求至少一个物理备库收到重做日志后，主库的事务才能够提交;

-3) 主库找不到合适的备库写入时，主库会自动关闭，防止未受保护的数据出现;

- 4) 优点: 该模式可以保证备库没有数据丢失:

- 5) 缺点: 主库的自动关闭会影响到主库的可用性，同时需要

备库恢复后才能提交，对网络等客观条件要求非常的高，主库的性能会因此受到非常大的冲击。

特点

* 零数据丢失，主备库保持一致
* 采用 SYNC 同步传输日志
* 备库需创建 standby redo logfiles
* 网络延迟或备库故障都会对主库带来性能影响
* 主库在阈值时间接收不到备库确认时，主库自动 shutdown。

### 最高可用 maximum availability

最大可用性模式

-1) 该模式提供了仅次于“最大保护模式”的数据保护能力;

-2) 要求至少一个物理备库收到重做日志后，主库的事务才能够提交;

-3) 主库找不到合适的备库写入时，主库不会关闭，而是临时降低到“最大性能模式”模式，直到问题得到处理;

- 4) 优点:该模式可以在没有问题出现的情况下，保证备库没有数据丢失，是一种折中的方法;

- 5) 缺点: 在正常运行的过程中缺点是主库的性能受到诸多因素的影响。

特点

* 采用 SYNC 同步传输日志；
* 备库需创建 standby redo logfiles；
* 网络延迟或备库故障都会对主库带来性能影响；
* 主库在阈值时间内接受不到备库确认时，会降级为最大性能模式，备库恢复正常后再恢复到最大可用模式。

### 最大性能 (默认) maximum performance

最大性能模式

1) 该模式是默认模式，可以保证主数据库的最高可用性;

1. 保证主库运行过程中不受备库的影响，主库事务正常提交，不因备库的任何问题影响到主库的运行;
2. 优点: 避免了备库对主数据库的性能和可用性影响:
3. 缺点: 如果与主库提交的事务相关的恢复数据没有发送到备库，这些事务数据将被丢失，不能保证数据无损失。

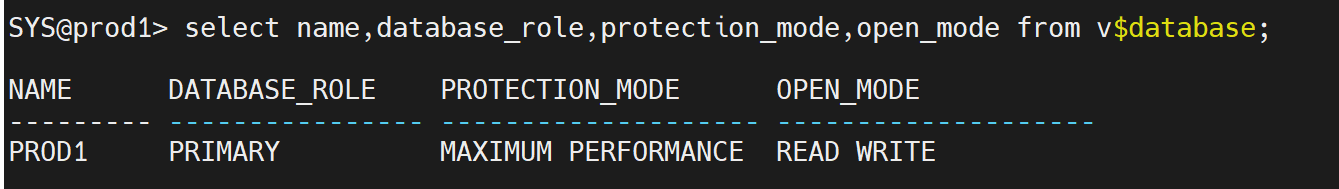
特点

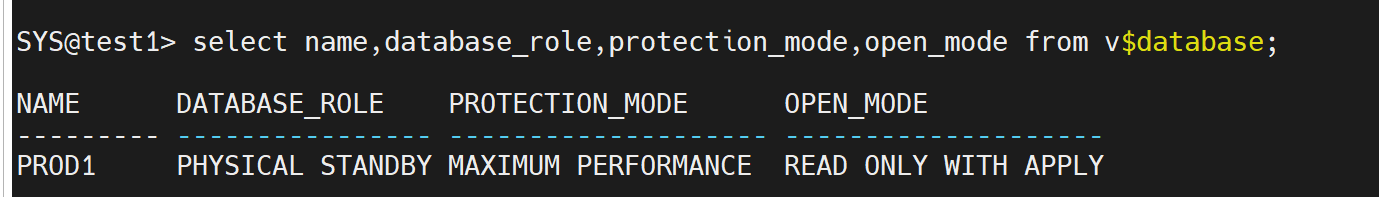
* 不能保证数据零丢失；
* 采用 async 异步传输日志；
* 网络和备库不会对主库带来性能影响。

## 相关命令

### 查看当前保护模式

select name,database\_role,protection\_mode,open\_mode from v$database;





## 创建 standby redolog

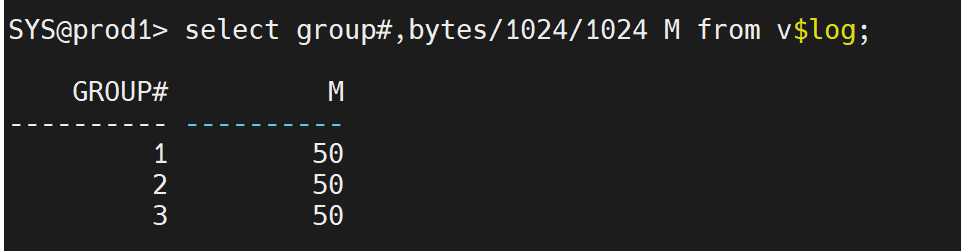
备库需要 standby redolog，standby redolog 至少比主库日志组多一组，日志文

件大小相同。

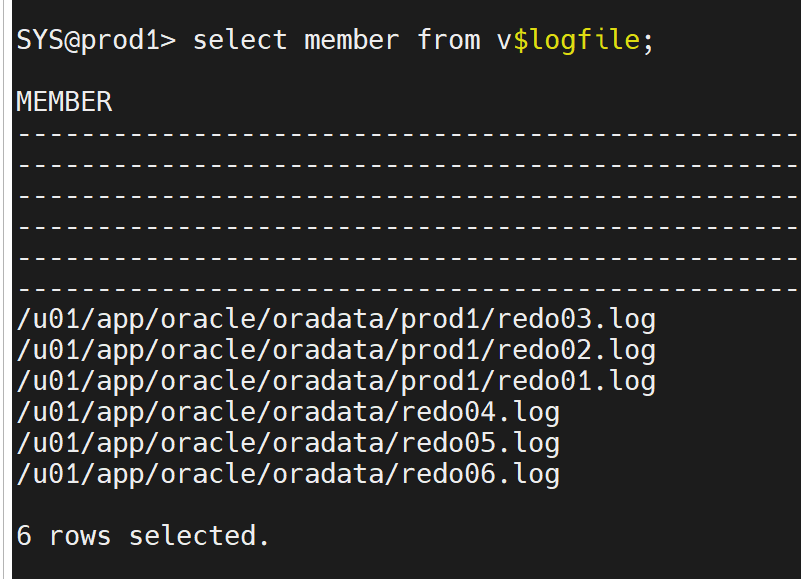
考虑到将来发生 switchover,在当前环境主库、备库都创建 standby redolog。

查看主库日志组

select group#,bytes/1024/1024 M from v$log;



select member from v$logfile;



### 主库添加 standby redolog

SYS@prod1>

alter database add standby logfile

'/u01/app/oracle/oradata/prod1/std\_redo01.log' size 50m;

alter database add standby logfile

'/u01/app/oracle/oradata/prod1/std\_redo02.log' size 50m;

alter database add standby logfile

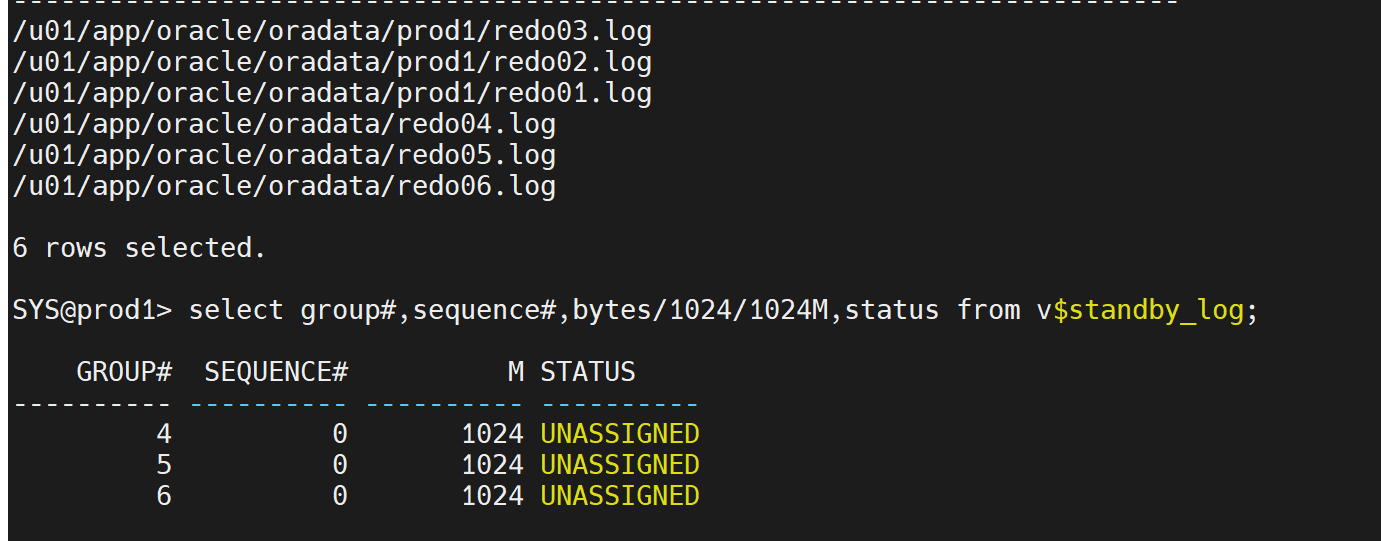
'/u01/app/oracle/oradata/prod1/std\_redo03.log' size 50m;

alter database add standby logfile

'/u01/app/oracle/oradata/prod1/std\_redo04.log' size 50m;

SYS@prod1>select member from v$logfile;

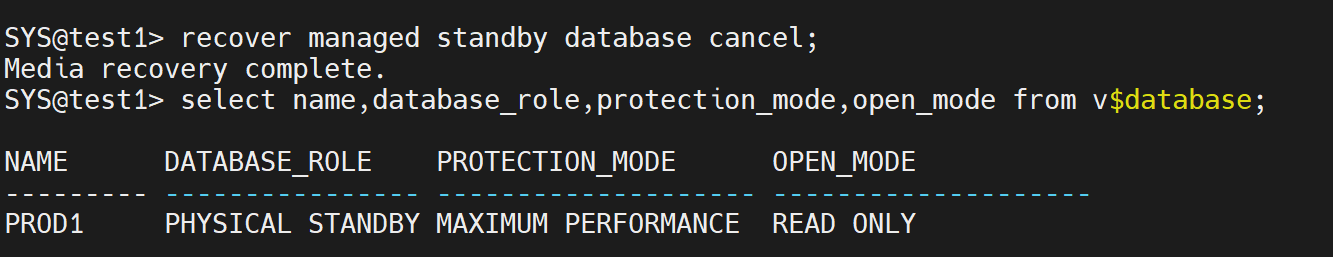
select group#,sequence#,bytes/1024/1024M,status from v$standby\_log;



### 备库取消日志应用

SYS@test1>recover managed standby database cancel;

select name,database\_role,protection\_mode,open\_mode from v$database;

****

### 备库创建 standby redolog

SYS@test1>alter database add standby logfile

'/u01/app/oracle/oradata/test1/std\_redo01.log' size 50m;

alter database add standby logfile

'/u01/app/oracle/oradata/test1/std\_redo02.log' size 50m;

alter database add standby logfile

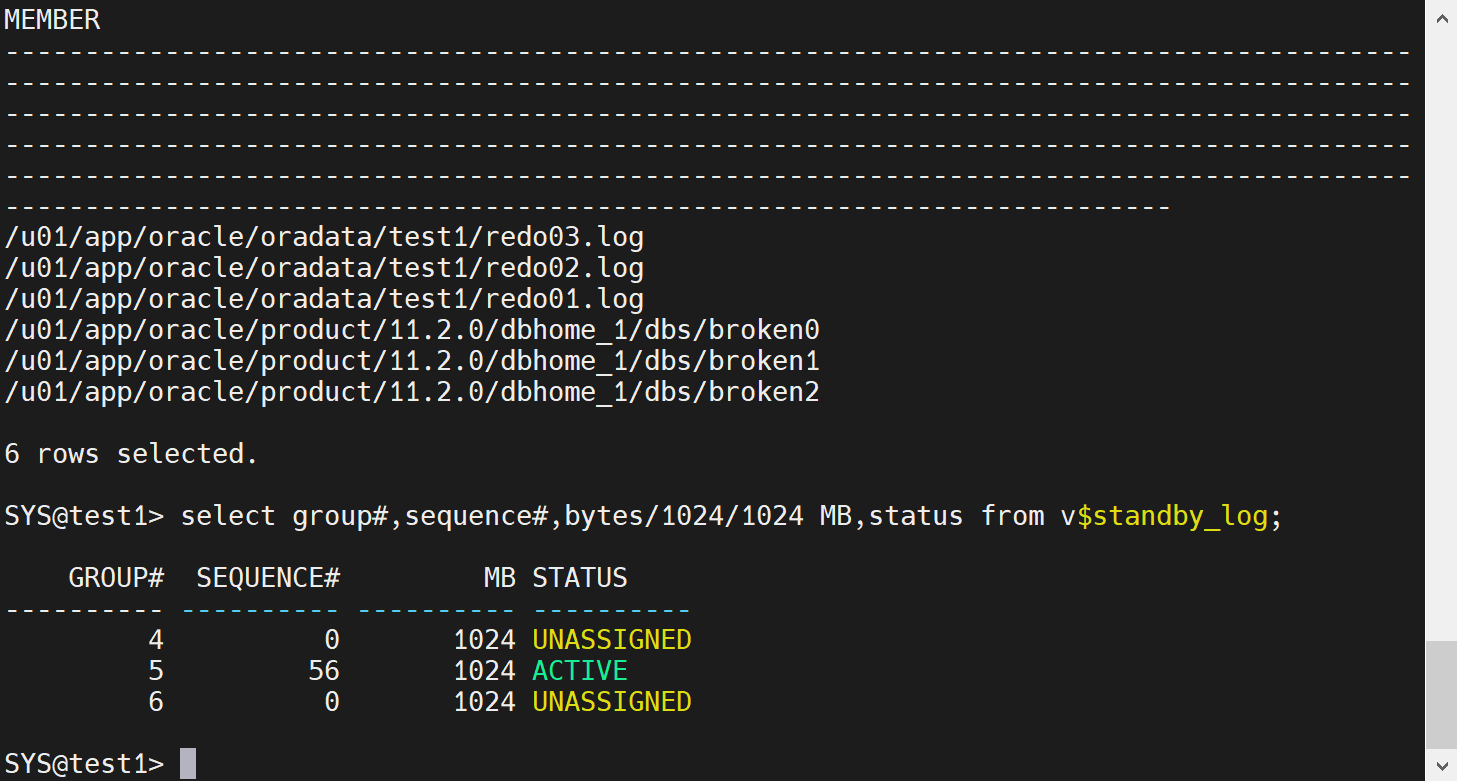
'/u01/app/oracle/oradata/test1/std\_redo03.log' size 50m;

alter database add standby logfile

'/u01/app/oracle/oradata/test1/std\_redo04.log' size 50m;

SYS@test1>select member from v$logfile;

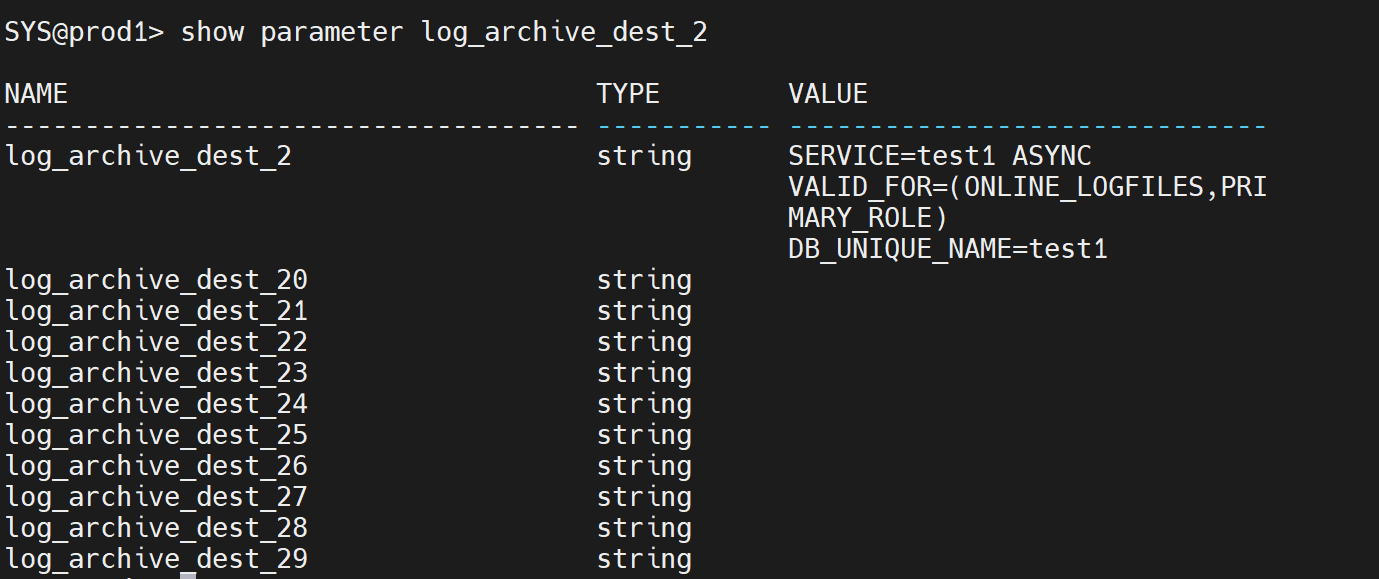
select group#,sequence#,bytes/1024/1024 MB,status from v$standby\_log;



### 修改日志传输模式为 sync

### 主库修改 log\_archive\_dest\_2 归档日志

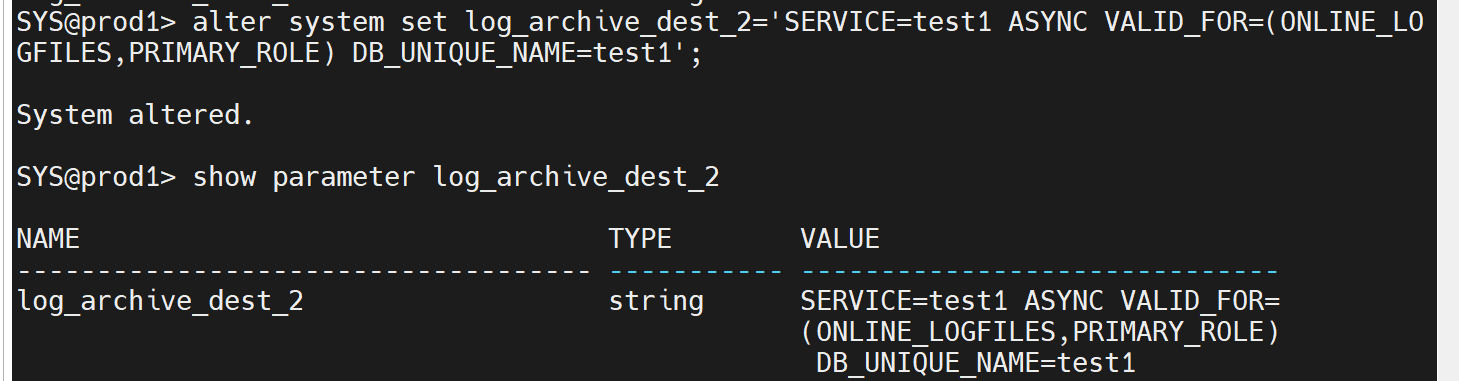
SYS@prod1>show parameter log\_archive\_dest\_2



SYS@prod1>alter system set log\_archive\_dest\_2='SERVICE=test1 SYNC

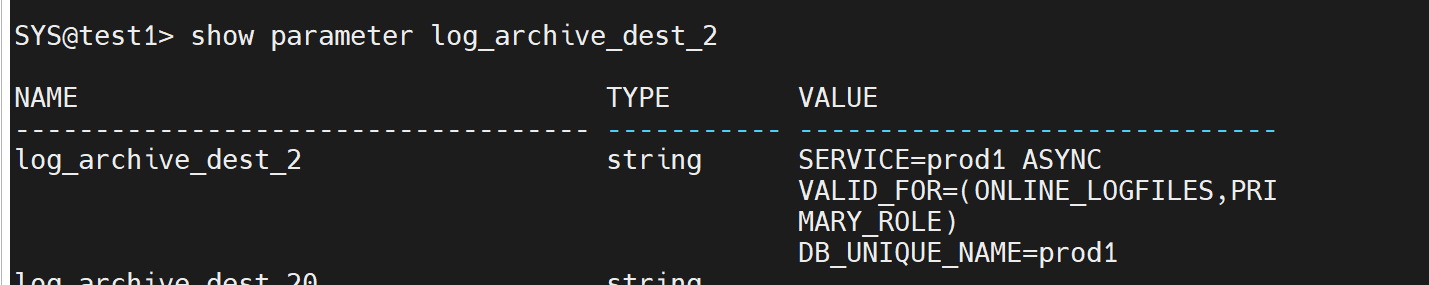
VALID\_FOR=(ONLINE\_LOGFILES,PRIMARY\_ROLE) DB\_UNIQUE\_NAME=test1';

SYS@prod1>show parameter log\_archive\_dest\_2



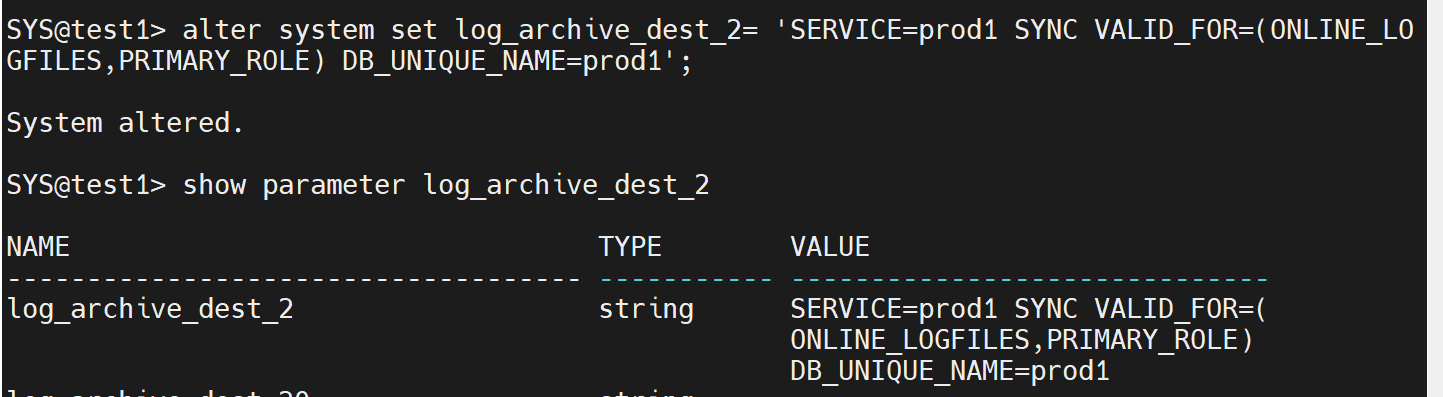
### 备库修改 log\_archive\_dest\_2

SYS@test1>show parameter log\_archive\_dest\_2



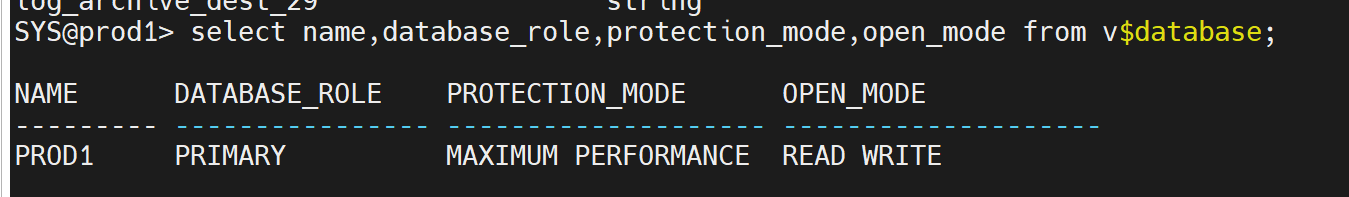
SYS@test1>alter system set log\_archive\_dest\_2= 'SERVICE=prod1 SYNC VALID\_FOR=(ONLINE\_LOGFILES,PRIMARY\_ROLE) DB\_UNIQUE\_NAME=prod1';

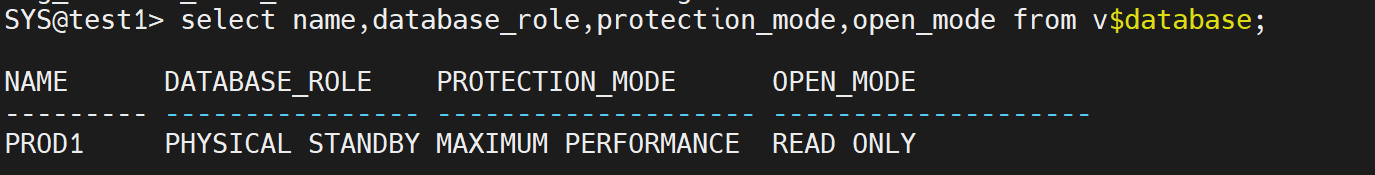
SYS@test1>show parameter log\_archive\_dest\_2



## 最大性能切最大保护

select name,database\_role,protection\_mode,open\_mode from v$database;

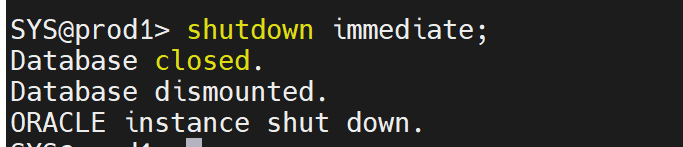




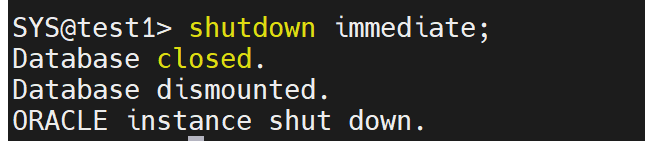
### 关库

最大性能切最大保护模式，需要关库 。先关主库，再关备库

SYS@prod1>shutdown immediate;

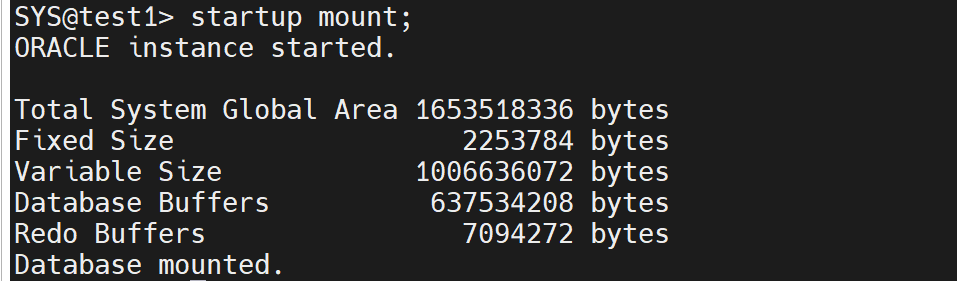


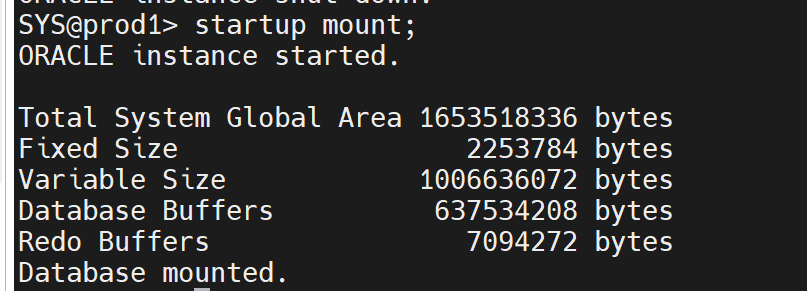
SYS@test1>shutdown immediate;



主备库启动到 mount 状态

startup mount;



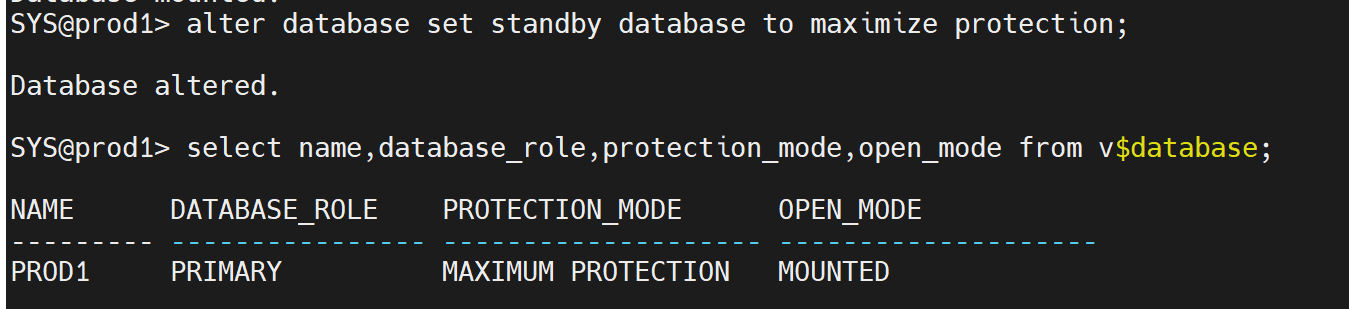


### 主库切最大保护

alter database set standby database to maximize performance;

SYS@prod1>alter database set standby database to maximize protection;

select name,database\_role,protection\_mode,open\_mode from v$database;



主库开库

SYS@prod1>alter database open;

select name,database\_role,protection\_mode,open\_mode from v$database;

备库查看保护模式

SYS@test1>select name,database\_role,protection\_mode,open\_mode from

v$database;

备库开库

SYS@test1>alter database open;

select name,database\_role,protection\_mode,open\_mode from v$database;

### 主库插数据测试

主库插数据，提交，切日志

SYS@prod1>

insert into scott.emp1 select \* from scott.emp1;

commit;

select count(\*) from scott.emp1;

alter system switch logfile;

备库应用日志

SYS@test1>recover managed standby database disconnect;

select name,database\_role,protection\_mode,open\_mode from v$database;

SYS@test1>select count(\*) from scott.emp1;

备库查看 standby 日志

SYS@test1>select group#,sequence#,bytes/1024/1024 MB,status from

v$standby\_log;

### 备库断开网络测试

[oracle@ocm2 ~]$ su - root

[root@ocm2 ~]# ifconfig

[root@ocm2 ~]# ifconfig eth0 down

主库插数据，commit 提交 hang

SYS@prod1>insert into scott.emp1 select \* from scott.emp1;

SYS@prod1>commit;

告警日志信息

备库恢复网络

[root@ocm2 ~]# ifconfig eth0 up

[root@ocm2 ~]# ifconfig

主库提交成功

超过阈值 关闭主库实例

如果主库长时间没有收到备库的确认，会 shutdown 掉主库实例。

备库端恢复网络

[root@ocm2 ~]# ifconfig eth0 up

重新启动主库

SYS@prod1>conn / as sysdba

SYS@prod1>startup

## 最大保护切最高可用

select name,database\_role,protection\_mode,open\_mode from v$database;

主库切最大可用模式

SYS@prod1>

alter database set standby database to maximize availability;

select name,database\_role,protection\_mode,open\_mode from v$database;

备库查看保护模式

SYS@test1>

select name,database\_role,protection\_mode,open\_mode from v$database;

# 实时应用 Real-Time Apply

默认备库只对 archived redo log 做 MRP 日志应用。

启用实时应用，备库可以对 standby redolog 进行日志应用。

需要备库创建 standby redolog。

## 启用 Real-Time Apply

主备库状态

select name,database\_role,protection\_mode,open\_mode from v$database;

## 备库取消日志应用

SYS@test1>recover managed standby database cancel;

select name,database\_role,protection\_mode,open\_mode from v$database;

## using current logfile 实时应用

SYS@test1>recover managed standby database using current logfile

disconnect;

select name,database\_role,protection\_mode,open\_mode from v$database;

告警日志

## 主库插入数据测试

SYS@prod1>insert into scott.emp1 select \* from scott.emp1;

SYS@prod1>commit;

SYS@prod1>select count(\*) from scott.emp1;

备库查询

SYS@test1>select count(\*) from scott.emp1;

# switchover 计划内主备库切换

"switchover"是指在Data Guard环境中将主数据库和备用数据库的角色进行转换。通常情况下，主数据库用于处理实时事务，而备用数据库用于提供灾难恢复和故障转移的能力。

当主数据库发生故障或需要维护时，可以执行"switchover"操作将备用数据库升级为主数据库的角色。这样，原来的主数据库将变为备用数据库，接管实时事务的处理。这种角色转换过程是无缝的，对应用程序和用户来说是透明的。

在执行"switchover"操作之前，需要确保主数据库和备用数据库之间的数据同步已经达到一定的程度，以避免数据丢失。在执行"switchover"操作后，主备数据库之间的数据同步将继续进行，以确保数据的一致性。

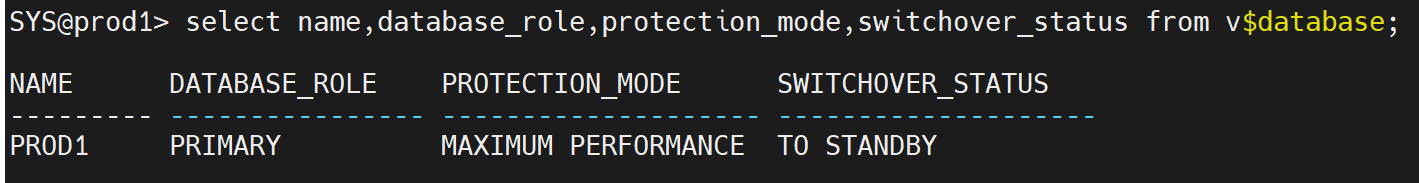
"switchover"操作可以通过Oracle Data Guard管理工具或命令行工具进行执行。执行"switchover"操作时，需要在备用数据库上执行一些预配置和准备工作，以确保角色转换的成功和数据的完整性。

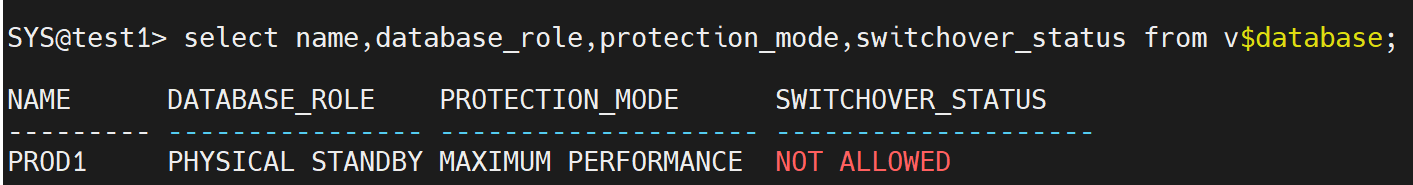
switchower:

## 查看主备库状态

SYS@prod1>

select name,database\_role,protection\_mode,switchover\_status from v$database;

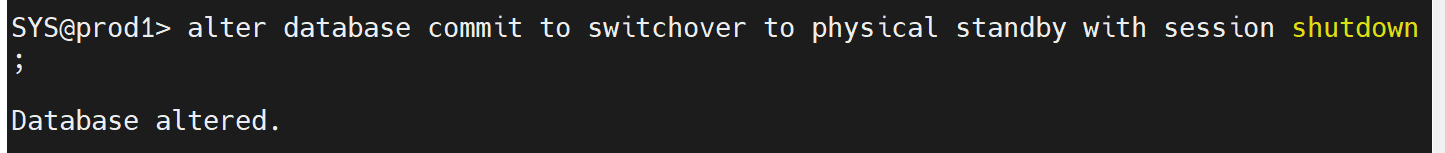




## 主库切物理备库

SYS@prod1>

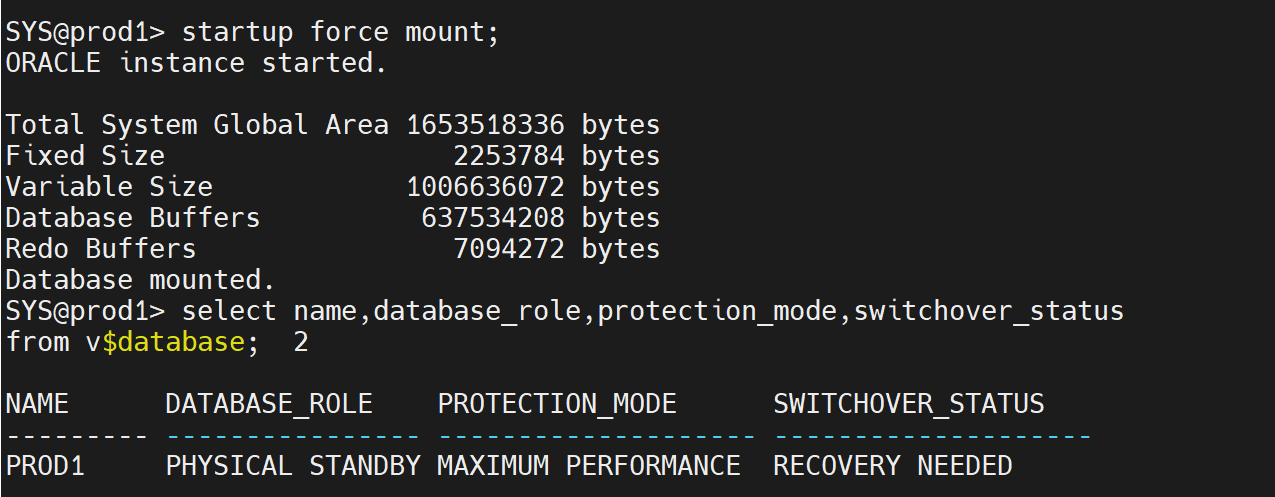
alter database commit to switchover to physical standby with session shutdown;



### 重启原来主库到 mount 状态

SYS@prod1>startup force mount;

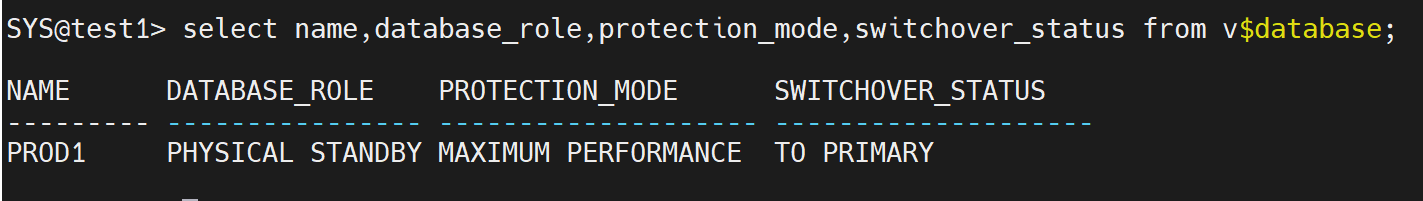
select name,database\_role,protection\_mode,switchover\_status from v$database;



### 查看原来备库状态

SYS@test1>

select name,database\_role,protection\_mode,switchover\_status from v$database;

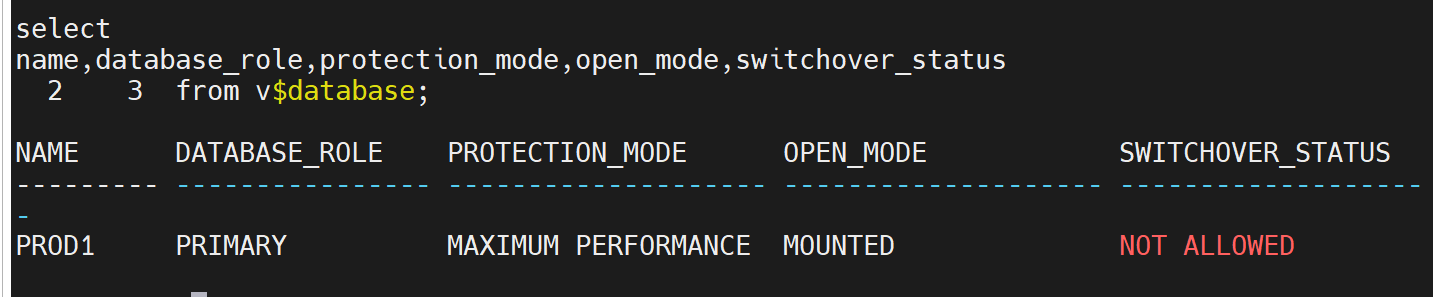


### 原备库切主库

SYS@test1>

alter database commit to switchover to primary with session shutdown;

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;

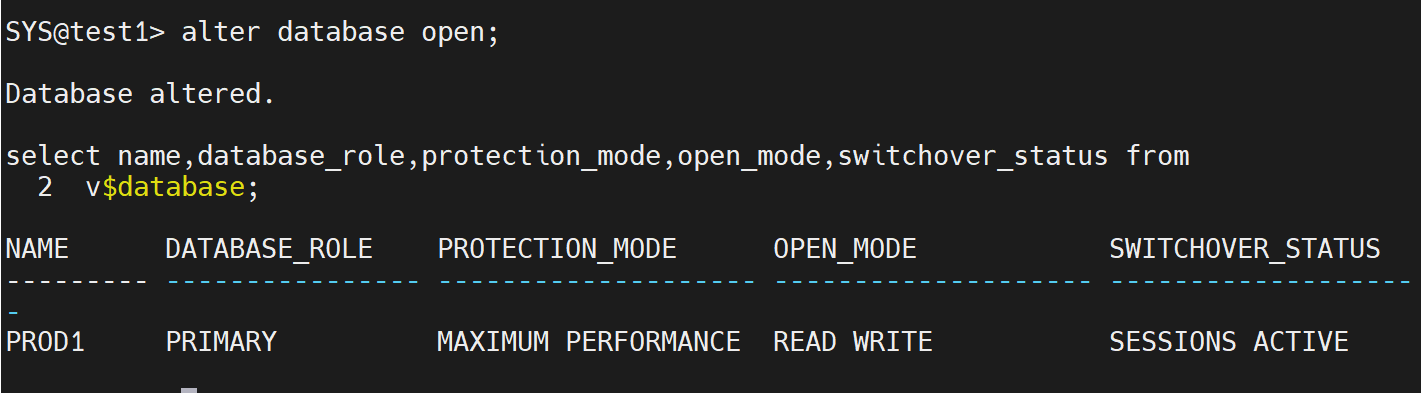


### open 打开新主库

SYS@test1>

alter database open;

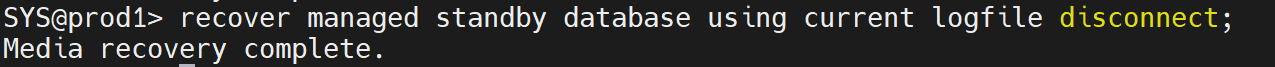
select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;



### 新备库应用日志

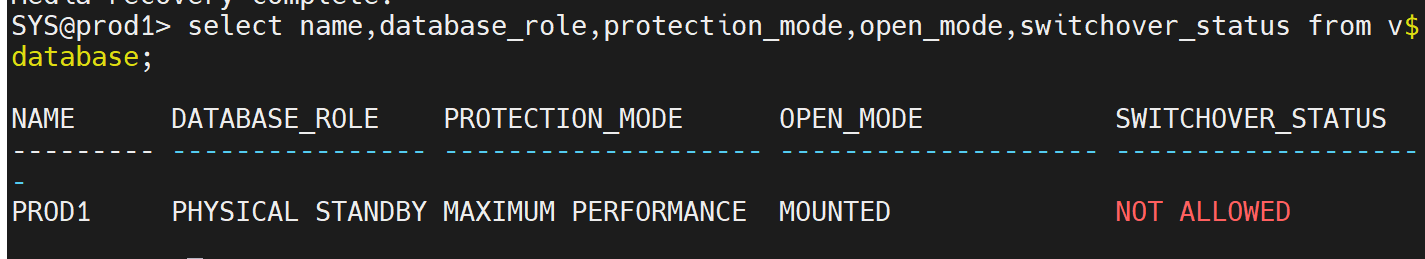
SYS@prod1>

recover managed standby database using current logfile disconnect;



SYS@prod1>

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;



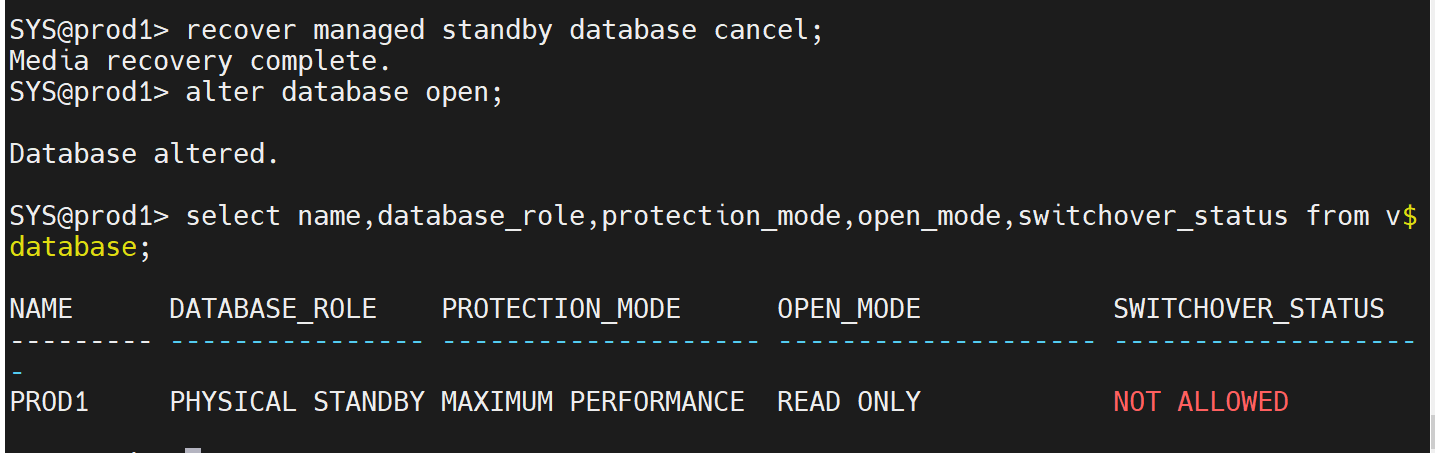
### 取消日志应用 open 打开备库

SYS@prod1>

recover managed standby database cancel;

alter database open;

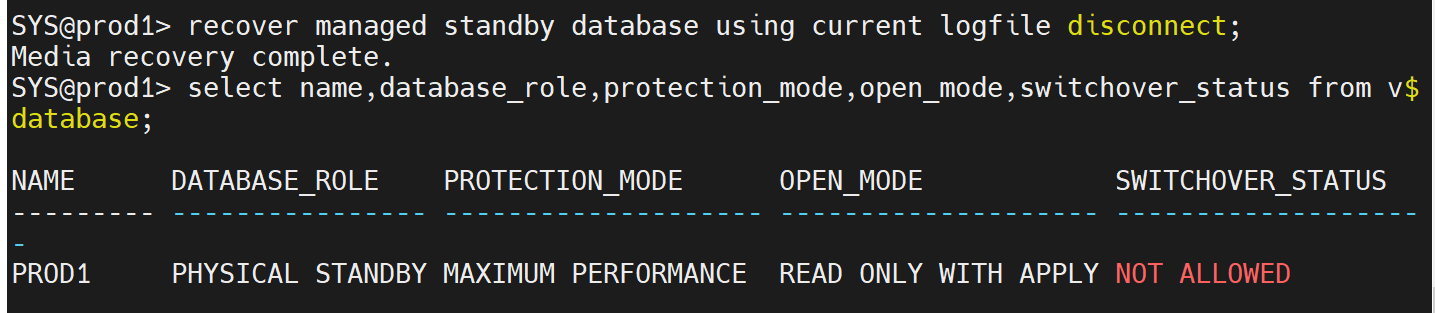
select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;



SYS@prod1>

recover managed standby database using current logfile disconnect;

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;



## 重新 switchover 主备切换

新主库状态

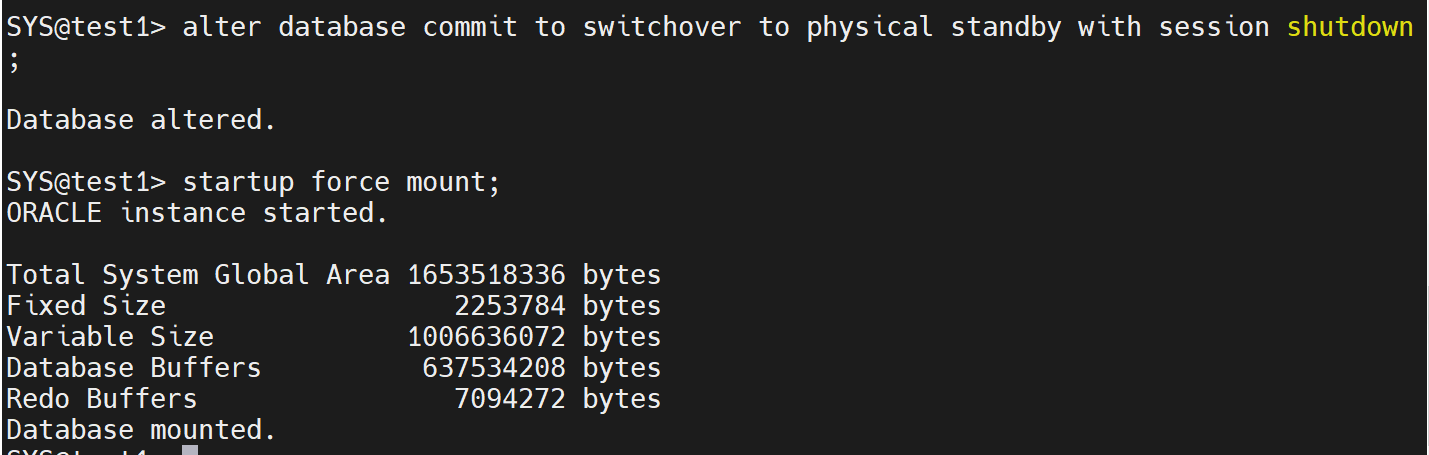
SYS@test1>

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;

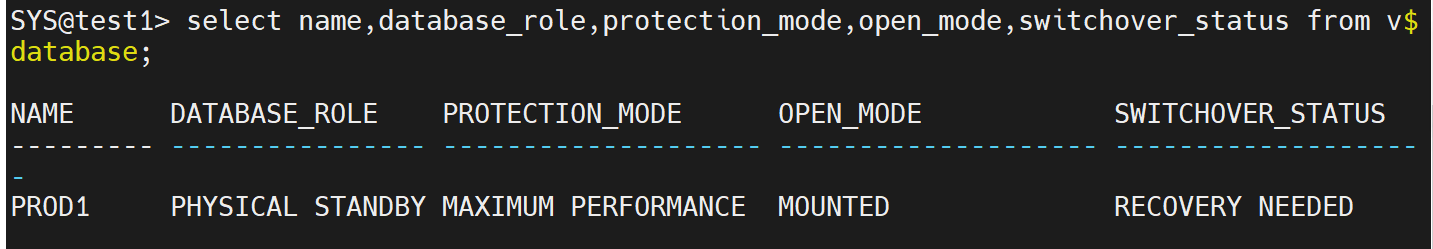
SYS@test1>

alter database commit to switchover to physical standby with session shutdown;

startup force mount;



select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;

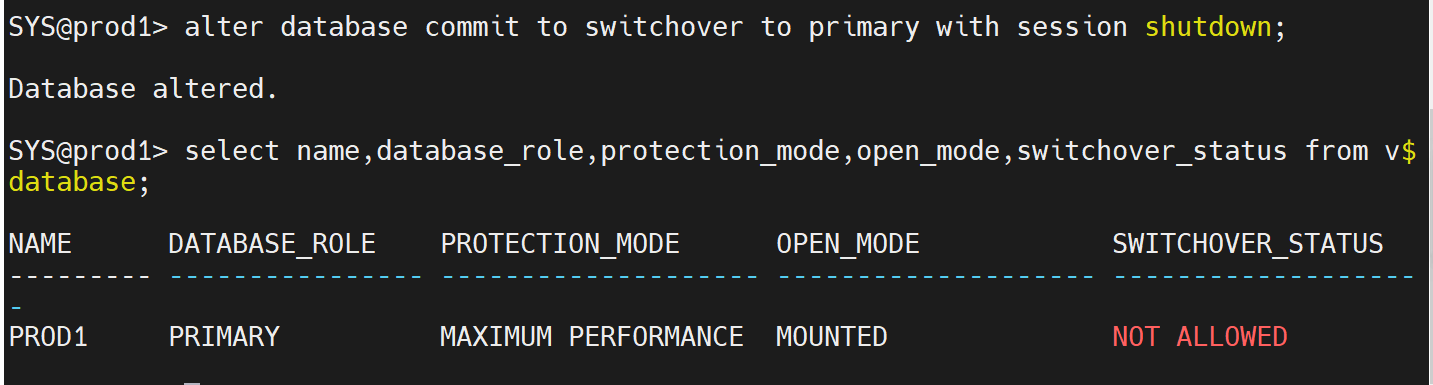


当前备库切主库

SYS@prod1>

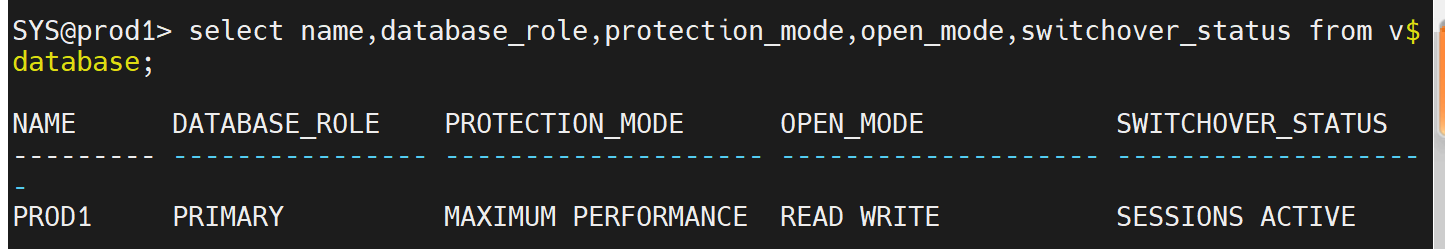
alter database commit to switchover to primary with session shutdown;

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;



alter database open;

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;

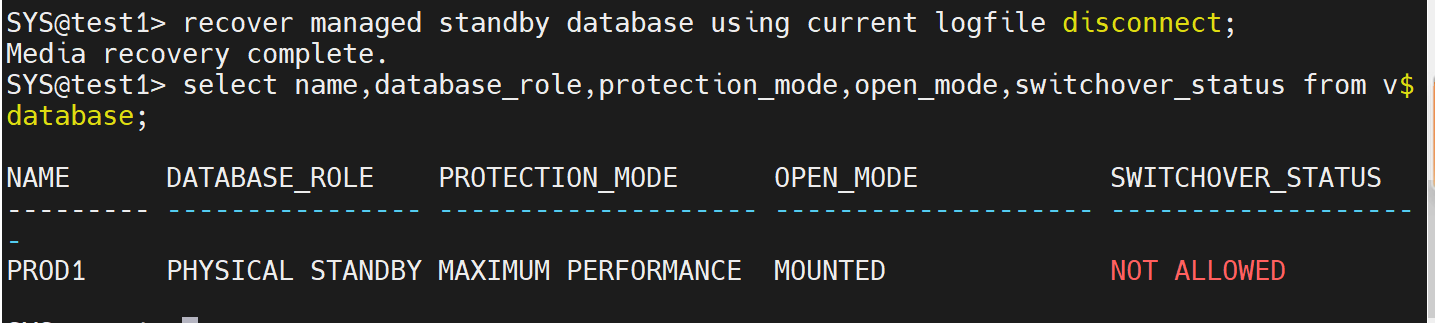


新备库应用日志

SYS@test1>

recover managed standby database using current logfile disconnect;

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;

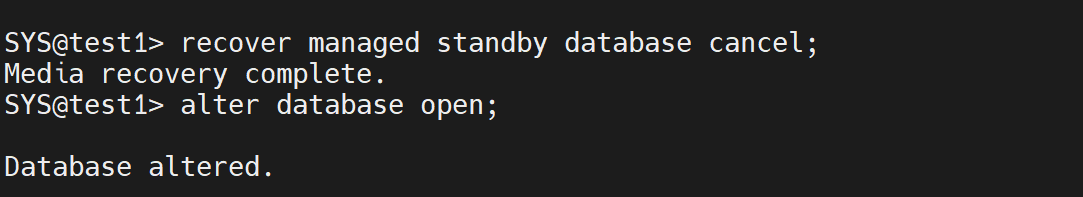


取消日志应用 开库

SYS@test1>

recover managed standby database cancel;

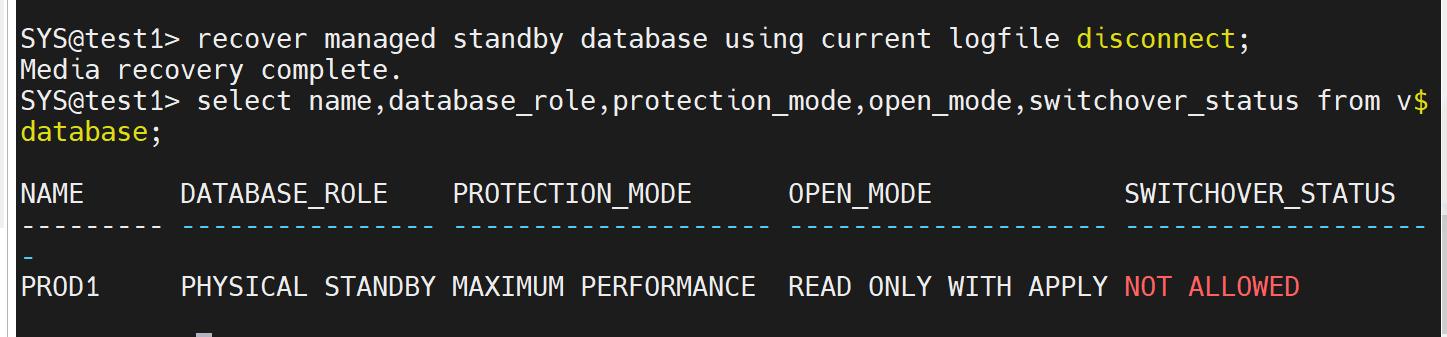
alter database open;



SYS@test1>

recover managed standby database using current logfile disconnect;

select name,database\_role,protection\_mode,open\_mode,switchover\_status from v$database;



# failover：主库意外宕机时，备库切换为主库

"failover"是指在Data Guard环境中将主数据库的角色自动切换到备用数据库的过程。当主数据库发生故障或不可用时，系统会自动将备用数据库提升为主数据库的角色，以确保业务的连续性和可用性。

Failover的过程是自动的，无需人工干预。当主数据库不可用时，Data Guard会检测到该情况，并自动触发failover操作。在failover过程中，备用数据库会接管主数据库的角色，成为新的主数据库，负责处理实时事务。

Failover操作需要确保主备数据库之间的数据同步已经达到一定的程度，以避免数据丢失。在执行failover操作后，原来的主数据库将变为备用数据库，而备用数据库将成为新的主数据库。

Failover操作可以通过Oracle Data Guard管理工具或命令行工具进行配置和执行。在配置failover时，需要设置自动故障转移（Automatic Failover）功能，并确保备用数据库具备足够的资源和性能来承担主数据库的角色。

## 配置 Fast-start failover

# 12c中DG新特性

Oracle Database 12c中的Data Guard（DG）引入了一些新的特性和增强功能，以提高数据保护和灾难恢复的能力。以下是一些12c中DG的新特性：

1. 多租户数据库的支持：12c中的Data Guard可以支持多租户数据库，这意味着可以在一个物理备库上保护多个租户的数据。这样可以减少硬件和管理成本，并提供更高的灵活性。

2. 快速故障转移：12c中引入了快速故障转移（Fast Failover）功能，可以在主库宕机时更快地将备库切换为新的主库。这减少了业务中断的时间，并提高了系统的可用性。

3. 自动重定向：12c中的Data Guard引入了自动重定向（Automatic Redirection）功能，可以在主库不可用时自动将客户端请求重定向到备库。这减少了手动干预的需要，并提高了系统的可用性。

4. 跨数据中心同步：12c中的Data Guard引入了远程同步（Far Sync）功能，可以在不同的数据中心之间实现实时的同步，以提供更高的灾难恢复能力。

5. 数据库压缩：12c中的Data Guard增加了对数据库压缩的支持，可以减少数据传输的带宽和存储需求。

6. 实时应用：12c中的Data Guard引入了实时应用（Real-Time Apply）功能，可以在备库上实时应用主库的日志，从而减少灾难恢复的时间。

7. 重建物理备库：12c中的Data Guard引入了重建物理备库（Recreate Physical Standby）功能，可以更方便地重建物理备库，以提高系统的可用性和可靠性。

# 参考文档

【说明】

以下列举了一些MOS的文章，可以在MOS上搜索Dataguard等关键字。该章学习期间，阅读或则模拟大量MOS文章中的内容和操作。

Step-By-Step Guide To Create Physical Standby On Normal File System For ASM Primary using RMAN (文档 ID 838828.1)

Step By Step Guide To Create Physical Standby Database Using RMAN Backup and Restore (文档 ID 469493.1)

Step By Step Guide On Creating Physical Standby Using RMAN Backup based Duplicate In ASM For ASM Primary (文档 ID 837102.1)

Step By Step Guide On How To Recreate Standby Control File When Datafiles Are On ASM And Using Oracle Managed Files (文档 ID 734862.1)

Data Guard Gap Detection and Resolution Possibilities (文档 ID 1537316.1)

Steps to perform for Rolling forward a standby database using RMAN Incremental Backup. (文档 ID 836986.1)

Data Guard Gap Detection and Resolution Possibilities (文档 ID 1537316.1)

11.2 Data Guard Physical Standby Switchover Best Practices using SQL\*Plus (文档 ID 1304939.1)

11gR2 RAC DB switchover using DG broker (文档 ID 880017.1)

11.2 Data Guard Physical Standby Switchover Best Practices using the Broker (文档 ID 1305019.1)