

在 Oracle RAC 系统上
安装和配置
Data Guard 11g

Oracle 白皮书
2008 年 10 月

最高
可用性
架构

Oracle 高可用性最佳实践

目录

1	引言	4
1.1	目的	4
1.2	背景	4
1.3	范围和应用	4
1.4	相关文档	5
1.5	前提	5
2	架构与概念	6
2.1	术语表	6
2.2	RAC 架构	6
2.3	Data Guard 架构	6
2.4	网络	7
2.5	Data Guard 架构	8
2.5.1	日志传输服务	8
2.5.2	日志应用服务	8
2.5.3	Data Guard Broker	9
2.6	工作原理	9
2.6.1	归档日志传送	9
2.6.2	备用重做日志	9
2.7	保护级别	10
2.7.1	最大保护	10
2.7.2	最高可用性	10
2.7.3	最高性能	10
2.7.4	优缺点	10
3	先决条件	11
3.1	硬件	11
3.2	网络	11
3.3	软件	11
3.4	Real Application Clusters	11
3.4.1	ASM	11
3.4.2	原始设备	11
4	创建 Data Guard 环境	13
4.1	前提	13
4.2	过程 – 概述	13
4.3	创建 RAC 备用数据库	14
4.3.1	配置主站点和备用站点	14
4.3.2	在每个站点上安装 Oracle 软件	14
4.3.3	服务器名称/VIP	14
4.3.4	配置 Oracle 网络	15
4.3.5	在每个节点上配置 ASM	17

4.3.6	准备主数据库以便进行复制.....	18
4.3.7	复制主数据库.....	20
4.3.8	为备用数据库创建 SPFILE.....	20
4.3.9	创建辅助控制文件.....	21
4.3.10	对备用数据库启用集群.....	22
4.3.11	临时文件.....	23
4.3.12	创建备用重做日志.....	23
4.4	使用 SQL Plus 配置 Data Guard.....	24
4.4.1	引言.....	24
4.4.2	配置备用数据库.....	24
4.4.3	配置主数据库.....	25
4.4.4	设置保护模式.....	25
4.4.5	启用重做传输和重做应用.....	26
4.5	使用 Data Guard Broker 配置 Data Guard.....	26
4.5.1	引言.....	26
4.5.2	Broker 配置文件.....	27
4.5.3	启用 Broker.....	27
4.5.4	创建 Broker 配置.....	27
4.5.5	启用 Broker 配置.....	27
4.5.6	Broker 自定义.....	28
5	监视.....	29
5.1.1	引言.....	29
5.1.2	日志文件.....	29
5.1.3	固定视图.....	29
6	管理.....	30
6.1	转换.....	30
6.1.1	使用 SQL Plus 执行转换.....	30
6.1.2	使用 Data Guard Broker 执行转换.....	30
6.2	故障切换.....	31
6.2.1	使用 SQL Plus 执行故障切换.....	31
6.2.2	使用 Data Guard Broker 执行故障切换.....	31
6.3	强制故障切换.....	31
6.3.1	使用 Data Guard Broker 执行强制故障切换.....	32
6.4	以只读方式打开备用数据库.....	32
6.5	实时应用/实时查询.....	32
7	附录 A – 使用 RMAN 创建备用数据库（传统方法）.....	33
7.1	前提.....	33
7.2	RMAN 备份.....	33
7.2.1	新备份.....	33
7.2.2	现有备份.....	34
7.3	创建备用数据库.....	34
7.3.1	先决条件.....	34
7.3.2	过程.....	34
8	附录 B – 其他阅读材料.....	36
8.1	Oracle 手册.....	36
8.2	Metalink.....	36

1 引言

1.1 目的

本文档介绍结合使用 Real Application Clusters (Oracle RAC) 11g 和 Data Guard 11g 来创建高可用性环境的整个过程。本文逐步介绍了包括基于 Oracle RAC 的主系统与物理备用系统的整个环境的构建过程，以此作为对标准文档的补充。Oracle 及其客户在研讨会和咨询工作中对这里介绍的过程进行了多次测试和运用，因而该过程体现了从这些实际工作中获得的实践经验。

1.2 背景

许多组织都在使用 Oracle 数据库来存储关键任务信息。即便在发生严重灾难的情况下，也要保证这类信息的安全。

实现这一目标的传统方法始终集中于 Oracle RAC 和面向硬件的解决方案，例如远程磁盘镜像。尽管 Oracle RAC 提供了高水平可用性，但通常所有 RAC 节点均位于同一机房内（除所谓的扩展集群之外）。Oracle RAC 集群提供了硬件层的高水平冗余，以确保不会出现单点故障。即便如此，在整个架构中仍然存在单点故障 — 单独的数据库或整个机房本身。如果机房遭到破坏或被隔离，那么很可能无法访问集群。如果数据库自身不可用（部分或全部毁坏），那么服务器级冗余不会有多大的帮助。

这正是实施灾难恢复计划的原因所在。该计划可包括将数据库恢复到另一台计算机的过程。但是，要完成这一恢复过程可能要花费数小时。更好的解决方案是配备一个恢复到一台独立计算机的影子数据库，该数据库始终保持最新。在发生灾难的情况下，可以使该数据库迅速处于联机状态。Oracle 提供的这一技术称为 Data Guard，而该数据库本身被称为备份数据库。尽管事实上 Data Guard 是一个涉及两种备用数据库（物理和逻辑）的概念性架构，但本文档的讨论只涉及物理备用数据库。

1.3 范围和应用

本文档将介绍一般情况下特别是在 Oracle RAC 环境中创建 Data Guard 安装所执行的步骤。本文档还将介绍一些具体过程，当您希望从主站点转换或故障切换到备用数据库所在站点时需要执行这些过程。

1.4 相关文档

- Oracle Data Guard 概念和管理 11g 第 1 版 2007 年 9 月
- Oracle Data Guard Broker 11g 第 1 版 2007 年 9 月

1.5 前提

本文中假设了下列前提：

- 生产系统与备用系统之间的网络连接满足保护级别要求及数据库重做信息生产量的传输要求
- 读者熟悉 Oracle Net 的配置
- 读者熟悉 Recovery Manager (RMAN)
- 读者熟悉 Oracle Real Application Clusters 的安装和配置

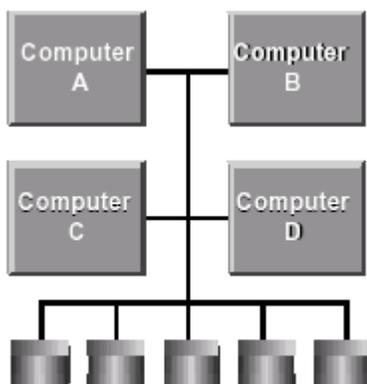
2 架构与概念

2.1 术语表

主站点	这是用户连接以访问生产数据库的站点。
备用站点	这是维护备用数据库的站点。也称作灾难恢复 (DR) 站点。在发生故障切换或转换的情况下用户将连接到该站点。
灾难	主站点不可用的情况。

2.2 RAC 架构

Oracle Real Application Clusters (Oracle RAC) 由数个单独的计算机系统组成，这些计算机通过高速互连接在一起，它们中的每一个都有权访问同一个磁盘组。



Oracle RAC 使用 ASM、RAW 分区或集群式文件系统在共享磁盘上创建一个数据库。Oracle 11g 中最常用的实现是使用 ASM。每个集群节点运行一个可以与该数据库交互的 Oracle 实例。

用户可以连接到这些 Oracle 实例中的任意一个来运行自己的应用程序。

2.3 Data Guard 架构

理想情况下，位于备用站点上的节点架构应与主站点上的节点架构相同，但这不是强制要求。

在规划备用系统架构时，需要考虑以下问题（在备用系统与主系统的架构不同时尤其如此）：

- 如果需要进行故障切换（主站点意外中断），备用站点能否处理预期负载？
- 如果在对主站点执行维护（计划中断）时要使用备用站点，备用站点能否处理预期负载？

假设已充分考虑了容量要求，那么以下就成为事实：不需要：

- 备用站点集群化。
- 备用站点使用 RAW 设备（除非它本身是集群）。

注意：现在已经不再严格要求两个站点（主站点与备用站点）都装备同样的软件版本和/或操作系统。客户现在可以有灵活的配置。当前所允许的配置的具体情况在下面的文档中提供：

Metalink [Note 413484.1](#):

“Data Guard 支持同一个 Data Guard 配置中的异构主系统和备用系统”

该说明将始终提供最新的支持信息。

注意：如果备用系统是集群，则该集群中只有一个节点能够与备用数据库同步。其余节点将维持空闲状态，直到备用系统上的数据库开启供常规（生产）运营使用。原始应用节点无论何种原因出现故障，Data Guard Broker（本文档稍后讨论）都会启用“应用节点”的自动故障切换，将应用节点自动切换到正常运行的备用集群节点。

2.4 网络

主站点与备用站点通过网络链接互相连接在一起。该网络链接必须可靠并且具有合适的带宽和延迟。

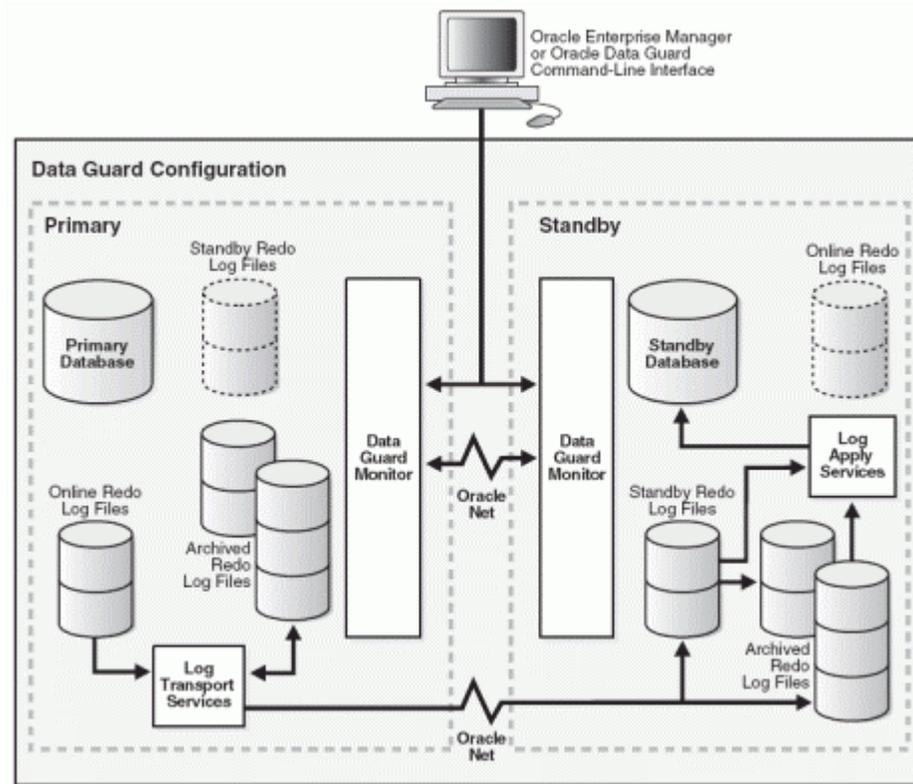
Data Guard 可提供 3 种不同的模式：

- 最大保护
- 最高可用性
- 最高性能

稍后将分别讨论这三种模式，但前 2 种模式需要同步写入到备用站点。注意，使用同步写入时通常存在性能折衷。尽管这些模式均提供了零数据丢失保护功能，但是，如果带宽受限，或者主数据库与备用数据库间的往返网络延迟 (RTT) 过大，则这些模式都会影响主数据库的性能。

2.5 Data Guard 架构

下图显示了 Data Guard 架构组件。下面会详细介绍这些组件。



2.5.1 日志传输服务

日志传输服务（又称为重做传输服务）旨在将主数据库的变更传播给备用数据库，其传播方式有二——传送归档日志 (ARCH)，或者在 Logwriter 进程处理重做数据的同时持续传送重做数据 (LNS)。

2.5.2 日志应用服务

日志应用服务（对物理备用数据库又称为“重做应用”，对逻辑备用数据库又称作“SQL应用”）负责将重做信息从归档重做日志文件或备用重做日志文件应用到备用数据库。

2.5.3 Data Guard Broker

Data Guard Broker 是一个管理和监视组件，它帮助创建、控制和监视由一个或多个物理备用数据库保护的主数据库。在 RAC 环境中支持使用 Broker。

2.6 工作原理

随着信息写入到主数据库，该信息也传输和应用到备用数据库。这些操作基本上分为两个阶段，这两个阶段之间并不直接互连。

2.6.1 归档日志传送

在这种情况下，主数据库实例将生成归档日志。归档日志一旦创建，就（由 ARCH 进程）立即传送到备用站点，然后在备用站点中将这些日志应用于数据库。可以立即执行应用操作，也可以在配置的延迟后再执行。

2.6.2 备用重做日志

传送重做信息的首选方法是利用 LNS 进程，随着 Logwriter 后台进程刷新重做缓冲区并写入联机重做日志文件，LNS 进程将会同时传输重做数据。这样就会通过网络持续地传输重做数据流，不会出现网络传输高峰。最佳实践要求接收重做数据的备用站点应配置有备用重做日志 (SRL)。于是传入的重做数据就直接写入这些日志。

SRL 的数量应该等于主数据库的所有联机重做日志的总数加 1，例如，如果有 4 个 RAC 实例，每个实例有 4 个重做日志，则应该配有 17 个 SRL。

每当向任何主实例的联机重做日志写入一个输入项，那么该输入项就同时写入其中一个备用重做日志。在对主联机重做日志执行日志切换操作时，备用数据库上也将发生一次日志切换，这就意味着当前的 SRL 将被归档到备用系统的一个本地目录。

Oracle 10g 中引入了“实时应用”特性。使用实时应用特性时，重做数据在写入到备用重做日志的同时就应用到备用数据库，而不是等待发生日志切换再应用重做。

2.7 保护级别

可以将 Data Guard 配置为提供不同程度的保护。这些不同程度的保护汇总如下：

2.7.1 最大保护

该解决方案可以 *确保* 在多种故障情况（如网络故障和主站点故障）下主站点与备用站点间不会出现数据丢失（零数据丢失）。如果最后一个备用站点无论因任何原因而不可用，则主数据库将停止工作。

2.7.2 最高可用性

该解决方案可以在单一故障情况下提供零数据丢失。然而，如果备用站点变为不可用（单一故障），该保护模式将重点放在“可用性”上，允许在主数据库上继续进行工作。如果在此期间发生第二个故障（例如，失去主数据库），则主数据库上的未受保护的事务将会丢失。一旦备用数据库恢复可用，无需人工干预，备用数据库会自动获取尚未传送的主数据库归档日志文件中的所有数据，并将再次重新同步。

2.7.3 最高性能

该解决方案有两种工作方式，一是通过 LNS 进程异步传送重做数据，二是在归档重做日志生成的同时通过 ARCH 进程将其从主数据库传送给备用数据库。这意味着如果发生故障，有可能造成一定程度的数据丢失，因为没有确保主数据库与备用数据库同步。

2.7.4 优缺点

最大保护和最高可用性模式要求主站点与备用站点之间有高规格的网络链接。随着数据写入到主数据库，它也同时写入到备用站点。这意味着，如果您的网络不具备所要求的带宽，或者存在 WAN 环境中常见的高 RTT 延迟，那么在考虑零数据丢失保护带来的更大好处的同时必须权衡对主数据库性能的潜在影响。

最高性能模式尽管对主数据库的性能没有影响，但其固有的异步重做传输可能在故障切换时造成数据丢失。

3 先决条件

3.1 硬件

理想情况下，备用站点将托管在能够支持主站点负载的硬件上。尽管这并非强制要求，但可以确保在需要故障切换到备用站点时，能获得同样的服务级别。

3.2 网络

为了帮助实现最大保护/最高可用性备用站点，主数据库应将重做日志信息同步地写入到主站点和备用站点。因此这些站点之间的网络链接必须：

- 可靠（无单点故障）。
- 具备合适的带宽（具体取决于预期的重做数据量）。
- 延迟很低。

3.3 软件

在以前的 Oracle 版本中，一直要求主站点与备用站点运行同一个版本的数据库软件。这一要求现在已宽松了很多。要了解当前支持哪些混合环境，请参阅[说明 413484.1](#)。

3.4 Real Application Clusters

3.4.1 ASM

如果主数据库使用 ASM 来进行磁盘存储，那么我们强烈建议您在备用站点上也配置 ASM。

注意：ASM 是 Oracle RAC 数据库的首选存储管理方式，并且现在非 RAC 数据库也在越来越多地使用这种存储管理方式。

3.4.2 原始设备

如果主数据库是 Oracle RAC 数据库且使用原始设备，并且备用数据库也是集群数据库，则备用站点上也需要配置原始设备。

但是，如果备用站点不是集群，则备用数据库无需使用原始设备。在这种情况下，以下配置将会简化管理：

3.4.2.1 非原始设备

1. 在备用站点创建一个存放数据文件的目录。
2. 在每个主 RAC 节点上创建同样的目录。
3. 在所有主 RAC 节点上创建从上述目录到原始设备的符号链接。
4. 创建 RAC 数据库时，使用这些符号链接而非原始设备名称。

优点

Data Guard 可以自动管理数据库文件。也就是说，如果有文件添加到主数据库，该文件也会自动添加到备用数据库。如果主站点与备用站点的目录结构不是完全相同，则必须使用 `init.ora` 参数 `db_file_name_convert` 和 `log_file_name_convert` 来配置文件名转换。

例如，无论哪个站点 `/u01/oradata/dbname/system.dbf` 都是相同的。

如果使用 Oracle 托管文件 (OMF)，也是这种情况。

4 创建 Data Guard 环境

4.1 前提

为了便于下文的讲述，我们特作如下假设：

- 两台主用主机的名称分别为 **europa** 和 **callisto**
- 两台备用主机的名称分别为 **dione** 和 **hyperion**
- 主数据库将称作 **MOON**
- 虚拟名称分别为 **europa-vip**、**callisto-vip**、**dione-vip** 和 **hyperion-vip**
- 主数据库和备用数据库均使用 ASM 进行存储
- 使用以下 ASM 磁盘组：**+DATA** 用于数据，**+FRA** 用于恢复/闪回
- 备用数据库将称作 **SUN**
- 将使用 Oracle 托管文件。
- ORACLE_BASE 设置为 **/u01/app/oracle**

下文用到这些名称的地方将像上面一样突出显示。

4.2 过程 — 概述

下面概述创建 Data Guard 环境的过程。后续各节将会详细介绍如何执行下面的每项任务：

1. 配置主站点和备用站点
2. 在每个站点上安装 Oracle 软件
3. 在每个站点上配置 Oracle 网络
4. 在两个站点上配置 ASM
5. 在每个站点上配置监听器
6. 在每个站点上配置 Oracle 网络
7. 创建初始化文件（主/备用）
8. 将主数据库复制到备用站点
9. 为备用站点创建服务器参数文件
10. 创建额外的备用控制文件
11. 创建备用重做日志文件
12. 将备用数据库注册到集群
13. 配置 Data Guard Broker
14. 将备用数据库置于适当的保护模式
15. 监视

4.3 创建 RAC 备用数据库

4.3.1 配置主站点和备用站点

为了简化环境的管理（以及 Data Guard 的配置），建议主计算机与备用计算机具有完全相同的目录结构，即

- 在两个站点上 ORACLE_HOME 指向相同的挂载点。
- 在两个站点上 ORACLE_BASE/admin 指向相同的挂载点。
- 在两个站点上 ASM 磁盘组相同。

4.3.2 在每个站点上安装 Oracle 软件

在两个站点上均从 Oracle 介质安装 Oracle 软件。安装的内容通常包括：

- Oracle Clusterware
- ASM 使用的 Oracle 数据库可执行文件
- RDBMS 使用的 Oracle 数据库可执行文件

4.3.3 服务器名称/VIP

在 Oracle Real Application Clusters 11g 中，由 Oracle Cluster Ready Services (CRS) 使用和维护虚拟服务器名称和 IP 地址。集群命名的示例如下：

注意：同时给出了短名称和完全限定名称。

服务器名称/别名/主机条目	作用
europa.local	公共主机名称（主节点 1）
callisto.local	公共主机名称（主节点 2）
dione.local	公共主机名称（备用节点 1）
hyperion.local	公共主机名称（备用节点 2）
europa-vip.local	公共虚拟名称（主节点 1）
callisto-vip.local	公共虚拟名称（主节点 2）

dione-vip.local	公共虚拟名称（备用节点 1）
hyperion-vip.local	公共虚拟名称（备用节点 2）

4.3.4 配置 Oracle 网络

4.3.4.1 在每个节点上配置监听器

每个节点都需要定义一个监听器，该监听器将从 ASM Oracle 主目录运行。在本示例配置中定义了如下监听器。

主角色
Listener_europa
Listener_callisto
Listener_dione
Listener_hyperion

4.3.4.2 静态注册

Oracle 必须能够访问两个数据库的所有实例，无论数据库是处于打开、挂载还是关闭状态。这意味着这些数据库必须静态注册到监听器。

这些注册项将有一个特定的名称，通过使用该名称将会方便 Data Guard Broker 的使用，稍后将对此进行讨论。

4.3.4.3 示例 Listener.ora

```

LISTENER_dione =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = dione-vip) (PORT = 1521)
          (IP = FIRST))
      )
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = TCP) (HOST = dione) (PORT = 1521)
          (IP = FIRST))
      )
      (ADDRESS_LIST =
        (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC))
      )
    )
  )
SID_LIST_LISTENER_dione =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME=SUN_dgmgrl.local)
      (SID_NAME = SUN1)
      (ORACLE_HOME = $ORACLE_HOME)
    )
  )

```

4.3.4.4 在每个站点上配置 TNS 条目

为使事情简单化，将在每个站点上同样地生成网络服务名称。这些服务名称分别称为：

别名	说明
MOON1_DGMGRL.local	使用服务名称 MOON_DGMGRL.local 指向 europa 上的 MOON 实例。这可以用来创建备用数据库。
MOON1.local	使用服务名称 指向 europa 上的 MOON 实例。
MOON2.local	使用服务名称 指向 callisto 上的 MOON 实例。
MOON.local	指向 MOON 数据库，即包含所有数据库实例。
SUN1_DGMGRL.local	使用服务名称 SUN1_DGMGRL 指向 dione 上的 SUN 实例。** 这将用于数据库复制。
SUN1.local	使用服务名称 SUN.local 指向 dione 上的 SUN 实例。
SUN2.local	使用服务名称 SUN.local 指向 hyperion 上的 SUN 实例。
SUN.local	指向 SUN 数据库，即包含所有数据库实例。
listener_DB_UNIQUE_NAME.local	这将是一个由两个地址行组成的 tns 别名条目。第一个地址行是节点 1 上的监听器的地址，第二个地址行是节点 2 上的监听器的地址。将上面两个监听器的地址放置在地址列表中，将会确保数据库自动注册到这两个节点。必须有两组条目。一组用于备用节点，名为 listener_SUN，一组用于主节点，名为 listener_MOON。

4.3.4.4.1 示例 tnsnames.ora (europa)

```
MOON1_DGMGRL.local =
  (DESCRIPTION =
    c (ADDRESS = (PROTOCOL = TCP) (HOST = europa-vip) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = MOON_DGMGRL.local)
    )
  )

MOON1.local =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = europa-vip) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = MOON.local)
      (INSTANCE_NAME = MOON1)
    )
  )

MOON2.local =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = callisto-vip) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = MOON.local)
    )
  )
```

```

        (INSTANCE_NAME = MOON2)
    )
)

MOON.local =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = europa-vip) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = callisto-vip) (PORT = 1521))
    (LOAD_BALANCE = yes)
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = MOON.local)
    )
  )

SUN1_DGMGRL.local =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = dione-vip) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = SUN_DGMGRL.local)
    )
  )

SUN2.local=
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = hyperion-vip) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = SUN.local)
      (INSTANCE_NAME=SUN2)
    )
  )

SUN1.local=
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = dione-vip) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = SUN.local)
      (INSTANCE_NAME=SUN1)
    )
  )

SUN.local=
  (DESCRIPTION =
    (ADDRESS_LIST=
      (ADDRESS = (PROTOCOL = TCP) (HOST = dione-vip) (PORT = 1521))
      (ADDRESS = (PROTOCOL = TCP) (HOST = hyperion-vip) (PORT = 1521)))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = SUN.local)
    )
  )

LISTENERS_MOON.local=
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP) (HOST = europa-vip) (PORT = 1521))
    (ADDRESS = (PROTOCOL = TCP) (HOST = callisto-vip) (PORT = 1521))
  )

```

4.3.5 在每个节点上配置 ASM

因为这是一个使用 ASM 的 Oracle RAC 数据库，因此强烈建议在进行后续操作前也在备用站点上配置 ASM。为简单起见，假定在备用站点上创建的磁盘组与主站点上的磁盘组具有相同的名称。

4.3.6 准备主数据库以便进行复制

有些初始化参数只有当数据库以备用数据库角色或者主数据库角色运行时才适用。如果在主站点和备用站点上对所有参数都进行了定义，就能确保如果发生角色转换（主用变备用，备用变新的主用），无需进行进一步的配置。

但是，有些参数是特定于节点的；因此需要分别定义两组参数，一组用于主站点节点，另一组用于备用站点节点。

4.3.6.1 主站点准备

在复制之前应该在主站点上设置如下初始化参数。尽管这些参数只适用于主站点，但在备用站点上也将同样地配置这些参数。

Dg_broker_config_file1	指向 ASM 磁盘组内的一个文件，注意无需存在该文件。
Dg_broker_config_file2	指向 ASM 磁盘组内的一个文件，注意无需存在该文件。
db_block_checksum	启用数据块完整性检查（可选）
db_block_checking	启用数据块一致性检查（可选）

如果对性能的影响在许可范围，并且不会违反现有的 SLA，就应强制启用 db_block_checksum 和 db_block_checking。

此外，还必须配置：

存档日志模式

主数据库必须置于存档日志模式。

强制日志记录

为了使备用数据库保持最新，需要在备用站点上应用记录于联机重做日志中的事务。在以前未使用 Data Guard 的一些环境中，可能使用了 NOLOGGING 选项来增强数据库性能。在 Data Guard 保护的环境中使用该特性极其不可取。

从 Oracle 9.2 版开始，Oracle 就引入了一个方法来防止出现 NOLOGGING 事务。该方法称作数据库的 *强制日志记录模式*。要启用强制日志记录，在主数据库上发出以下命令：

```
alter database force logging;
```

口令文件

必须将主数据库配置为使用外部口令文件。该配置通常是在安装时进行的。如果安装时未进行该配置，可以使用以下命令创建口令文件：

```
orapwd file=$ORACLE_HOME/dbs/orapwMOON1 password=myspasswd
```

发出该命令前请确保 ORACLE_SID 设置为相应的实例 — 在该示例中为 MOON1。

对集群的每个节点重复这一操作。

此外，还要确保初始化参数 `remote_login_passwordfile` 设置为 “exclusive”。

对于 Oracle11.1，可以选择通过 SSL 对重做传输的 Oracle Net 对话进行验证（参阅 “Data Guard 概念手册” 的 6.2.1 节）。

4.3.6.2 备用站点准备

初始化文件

作为复制过程的一部分，将使用一个临时的初始化文件。在本文档的讨论范围内，该文件名为 `/tmp/initMOON.ora`，它含有一行：

```
db_name=MOON
```

口令文件

必须将备用数据库配置为使用口令文件。必须从主站点复制口令文件到备用站点并对其重命名以体现备用实例来创建该文件。

对集群的每个节点重复这一操作。

此外，还要确保初始化参数 `remote_login_passwordfile` 设置为 “exclusive”。

创建审计文件目标

在备用系统的每个节点上创建一个目录来存放审计文件。

```
mkdir /u01/app/oracle/admin/SUN/adump
```

启动备用实例

现在所有事情已经就位，需要启动备用实例来准备开始复制：

```
export ORACLE_SID=SUN1
sqlplus / as sysdba
startup nomount pfile='/tmp/initMOON.ora'
```

测试连接

使用以下命令测试主数据库到备用数据库的连接：

```
sqlplus sys/myspasswd@SUN_dgmgr1 as sysdba
```

其结果应该是成功连接。

4.3.7 复制主数据库

备用数据库通过主数据库来创建。对于 Oracle10g 及之前的版本，这一过程需要制作一个主数据库备份，并将其传给备用数据库，然后再进行恢复。Oracle RMAN 11g 通过一种新方法简化了这一过程，这种方法允许执行“动态”复制。本文将使用这种方法（11g 之前的方法在附录中介绍）。

使用以下命令从主数据库调用 RMAN:

```
export ORACLE_SID=MOON1
rman target / auxiliary sys/mypasswd@SUN1_dgmgr1
```

注意: 如果 RMAN 返回错误“rman: can't open target”，则要确保 PATH 的开头存在“ORACLE_HOME/bin”，因为已经有一个名为 RMAN 的 Linux 实用程序。

接着，发出以下复制命令:

```
duplicate target database for standby from active database
spfile
set db_unique_name='SUN'
set control_files='+DATA/SUN/controlfile/control01.dbf'
set instance_number='1'
set audit_file_dest='/u01/app/oracle/admin/SUN/adump'
set remote_listener='LISTENERS_SUN'
nofilenamecheck;
```

4.3.8 为备用数据库创建 SPFILE

默认情况下，RMAN 复制命令将为位于 \$ORACLE_HOME/dbs 中的实例创建 spfile。

该文件将包含引用主数据库上的实例名称的条目。作为该创建过程的一部分，会更改数据库名称以反映备用数据库的 DB_UNIQUE_NAME，因此创建的 spfile 基本无用。现在，将使用主数据库的 spfile 的内容来创建新的 spfile。

4.3.8.1 获取控制文件的位置

开始本过程前，请从当前运行的备用数据库记下 control_files 参数的值。

4.3.8.2 创建文本初始化文件

该过程的第一步需要将主数据库的初始化参数转储为一个文本文件:

```
set ORACLE_SID=MOON1
```

```
sqlplus "/ as sysdba"
create pfile='/tmp/initSUN.ora' from spfile;
```

将创建的“/tmp/initSUN.ora”文件复制到备用服务器。

4.3.8.3 编辑 init.ora

在备用服务器上，编辑 /tmp/initSUN.ora 文件：

注意：将每个出现的 MOON 更改为 SUN，DB_NAME 参数除外，该参数一定**不要**更改。

设置 control_files 参数以反映上面 4.3.8.1 中获得的值。这最有可能是 +DATA/SUN/controlfile/control01.dbf。

保存更改。

4.3.8.4 创建 SPFILE

前面创建了文本初始化文件，现在需要发出以下命令将其转换为 spfile 并存储在 ASM 内：

```
export ORACLE_SID=SUN1
sqlplus "/ as sysdba"
create spfile='+DATA/SUN/spfileSUN.ora' from pfile=
'/tmp/initSUN.ora'
```

4.3.8.5 创建指针文件

现在 ASM 中已经有了 spfile，需要告诉 RDBMS 实例在何处可以找到该文件。

在备用节点 1 (dione) 的 \$ORACLE_HOME/dbs 目录中创建名为 initSUN1.ora 的文件。该文件只有一行内容：

```
spfile='+DATA/SUN/spfileSUN.ora'
```

在备用节点 2 (hyperion) 的 \$ORACLE_HOME/dbs 目录中创建名为 initSUN2.ora 的文件。该文件也只有一行内容：

```
spfile=' +DATA/SUN/spfileSUN.ora'
```

另外，从备用节点 1 (dione) 的 \$ORACLE_HOME/dbs 中删除 RMAN 创建的 spfile

4.3.9 创建辅助控制文件

RMAN 复制完成后，它创建的备用数据库只有一个控制文件。这不是最佳实践，因此该过程的下一步是创建额外的控制文件。

该过程有两个阶段：

1. 关闭数据库并使用 nomount 启动它：

```
shutdown immediate;
```

```
startup nomount;
```

2. 将 control_files 参数的值改为 '+DATA',' +FRA'

```
alter system set control_files='+DATA',' +FRA' scope=spfile;
```

3. 关闭数据库并重新启动它:

```
shutdown immediate;  
startup nomount;
```

3. 使用 RMAN 复制已经存在的控制文件:

```
export ORACLE_SID=SUN1  
rman target /  
restore controlfile from '+DATA/SUN/controlfile/control01.dbf'
```

该操作将同时在 ASM 磁盘组的 +DATA 和 +FRA 中创建控制文件。这也将更新 spfile 中的控制文件参数。

如果希望有 3 个控制文件，只需更新 control_files 参数，使其包含原始控制文件以及刚创建的两个控制文件。

4.3.10 对备用数据库启用集群

现在需要将备用数据库置于集群件的控制之下，即将其注册到 Cluster Ready Services。

在开始之前，请检查是否可以在第二个备用节点 (hyperion) 上启动实例:

```
export ORACLE_SID=SUN2  
sqlplus "/ as sysdba"  
startup mount;
```

注意：在进行后续步骤前，请解决所有问题。

4.3.10.1 确保配置了服务器端负载均衡

请检查在备用实例中是否定义了 init.ora 参数 remote_listener。

如果该参数不存在，则在（所有备用节点的）tnsnames.ora 文件中以下面的格式创建一个条目:

```
LISTENERS_SUN.local =  
  (DESCRIPTION =  
    (ADDRESS_LIST =  
      (ADDRESS = (PROTOCOL = TCP) (HOST = dione-vip.local) (PORT = 1521))  
      (ADDRESS = (PROTOCOL = TCP) (HOST = hyperion-vip.local) (PORT =  
1521))  
    )  
  )
```

然后将参数 `remote_listener` 的值设为 `LISTENERS_SUN.local`。

4.3.10.2 将数据库注册到 CRS

发出以下命令将数据库注册到 Oracle Cluster Ready Services:

```
srvctl add database -d SUN -o $ORACLE_HOME -m local -p
"+DATA/SUN/spfileSUN.ora" -n MOON -r physical_standby -s mount
srvctl add instance -d SUN -i SUN1 -n dione
srvctl add instance -d SUN -i SUN2 -n hyperion
```

4.3.10.3 测试

停止所有正在运行的实例，然后使用以下命令启动数据库（所有实例），以此测试上述操作生效:

```
srvctl start database -d SUN
```

一旦启动，使用以下命令检查相关实例是否正在运行:

```
srvctl status database -d SUN
```

4.3.11 临时文件

对于备用数据库来说，临时表空间的相关临时文件是自动创建的。

4.3.12 创建备用重做日志

当使用 Logwriter (LGWR) 配置传输时（这是默认配置），需要使用备用重做日志 (SRL) 存储来自主数据库的重做数据。

每个备用重做日志文件至少要与主数据库中最大的重做日志文件一样大。建议使主数据库中的所有重做日志文件与相应的备用数据库中的备用重做日志具有相同的大小。

建议的 SRL 个数为:

(每个主实例的联机重做日志数 + 1) * 实例数。

尽管备用重做日志仅由备用站点使用，但是在主站点和备用站点上都应定义备用重做日志。这将确保在主备数据库转换角色时（主用变为备用，备用变为主用），无需执行额外的配置。

必须先挂载备用数据库（默认情况下作为“备用”挂载），才能创建 SRL。

执行以下操作创建 SRL（下面给出的大小只是示例，必须针对当前环境进行调整）:

```
1. sqlplus '/ a sysdba'
2. startup mount
```

```
3. alter database add standby logfile SIZE 100M;
```

注意：备用重做日志也是在 logfile 组中创建的。但是要知道这样的事实：备用重做日志的组编号必须大于与主数据库中的 ORL 关联的组的编号。Oracle 的 Wrt 组编号在 ORL 和 SRL 之间没有差别。

注意：在主数据库和备用数据库上都需要创建备用重做日志。

现在已经创建了备用数据库。该过程的下个阶段主要是启用事务同步。有两种方法可以实现这一目的：

1. 使用 SQL Plus
2. 使用 Data Guard Broker

4.4 使用 SQL Plus 配置 Data Guard

4.4.1 引言

本节介绍使用 SQLPlus 设置物理备用数据库环境以及手动设置数据库初始化参数的过程。

4.4.2 配置备用数据库

在备用数据库上需要设置以下初始化参数：

参数	值 (dione)	值 (hyperion)
db_unique_name	SUN	
db_block_checking	TRUE (可选)	
db_block_checksum	TRUE (可选)	
log_archive_config	dg_config=(MOON, SUN)	
log_archive_max_processes	5	
fal_client	SUN1.local	SUN2.local
fal_server	'MOON1.local', 'MOON2.local'	
Standby_file_management	Auto	
log_archive_dest_2	service=MOON LGWR SYNC AFFIRM db_unique_name=PRIMARY_MOON VALID_FOR=(ALL_LOGFILES,PRIMARY_ROLE)	

log_archive_dest_2 (Max. Performance Mode)	service=MOON ARCH db_unique_name=PRIMARY_MOON VALID_FOR=(ALL_LOGFILES,PRIMARY_ROLE)
---	---

4.4.3 配置主数据库

在主数据库上需要设置以下初始化参数：

参数	值 (europa)	值 (callisto)
db_unique_name	MOON	
db_block_checking	TRUE (可选)	
db_block_checksum	TRUE (可选)	
log_archive_config	dg_config=(MOON, SUN)	
log_archive_max_processes	5	
fal_client	MOON1.local	MOON2.local
fal_server	'SUN1.local', 'SUN2.local'	
standby_file_management	Auto	
Log_archive_dest_2	service=SUN LGWR SYNC AFFIRM db_unique_name=SUN VALID_FOR=(ALL_LOGFILES,PRIMARY_ROLE)	
Log_archive_dest_2 (Max. Performance Mode)	service=SUN ARCH db_unique_name=SUN VALID_FOR=(ALL_LOGFILES,PRIMARY_ROLE)	

4.4.4 设置保护模式

为了指定保护模式，必须挂载主数据库，但不打开它。

注意：数据库必须以独占方式挂载，该模式实际意味着，只启动一个实例且其参数设置为 cluster_database=false，除此之外，其他所有 RAC 实例均关闭。

一旦处于这种情况，就要在主站点上发出以下语句：如果使用最大保护模式，则使用

命令：

```
Alter database set standby database to maximize protection;
```

如果使用最高可用性模式，则使用命令：

```
Alter database set standby database to maximize availability;
```

如果使用最高性能模式，则使用命令：

```
Alter database set standby database to maximize performance;
```

4.4.5 启用重做传输和重做应用

通过以下操作启用对备用数据库的重做传输和重做应用：

4.4.5.1 备用站点

需要将备用数据库置于托管恢复模式。为此发出以下语句：

```
Alter database recover managed standby database disconnect;
```

Oracle 10gR2 引入了实时重做应用（需要 SRL）。通过发出以下语句来启用实时应用：

```
alter database recover managed standby database using current  
logfile disconnect;
```

4.4.5.2 主站点：

在 init.ora 文件中设置：

```
log_archive_dest_state_2=enable
```

或通过 SQLPlus 发出：

```
alter system set log_archive_dest_state_2=enable
```

4.5 使用 Data Guard Broker 配置 Data Guard

4.5.1 引言

Data Guard Broker 有一个命令行界面，可用于简化整个 Data Guard 环境的管理。使用 Broker 时，配置信息存储在 Broker 本身。Broker 启动时（通过数据库初始化参数启用），它将使用一系列 ALTER SYSTEM 语句来设置主站点和备用站点之间的事务同步。它设置的参数与上述 SQLPlus 示例中提到的参数相同。**因此必须通过 Broker 更改数据库配置**，而不能通过直接编辑初始化参数来更改。不遵守这一原则将导致 Broker 覆盖这些值。

注意：如果使用 Grid Control 管理 Data Guard 环境，则必须配置 Broker。

4.5.2 Broker 配置文件

Data Guard Broker 使用两个文件存放其配置信息。默认情况下，这两个文件均位于 `$ORACLE_HOME/dbs` 目录中。在 RAC 环境中，这样放置文件不合适，因为所有数据库实例都需要能够访问同一个配置信息。

在继续进行 Broker 配置之前，请确保将 Broker 文件配置为指向共享存储（本案中为 ASM）。

可以查看以下参数值来确认这一点：`dg_broker_config_file1` 和 `dg_broker_config_file2`。在上文 (4.3.6.1) 中可以看到这些参数的合适值。

4.5.3 启用 Broker

必须首先启用 Broker 才能使用它。为此，将数据库初始化参数 `dg_broker_start` 的值改为 `true`：

```
alter system set dg_broker_start=true;
```

注意：在主站点和备用站点上都要执行这一操作。

4.5.4 创建 Broker 配置

可以使用 Grid Control 或 DGMGRL 命令行界面来创建 Broker 配置。该文档中使用后者。

使用以下命令启动 `dgmgrl`

```
dgmgrl sys/mypasswd
```

** 注意：不要单独使用 `/`，因为这样做可能导致以后出现问题。输入以

下命令来创建 Data Guard 配置：

```
create configuration MOON_SUN as primary database is MOON
connect identifier is MOON.local;
```

```
add database SUN as connect identifier is SUN.local
maintained as physical;
```

4.5.5 启用 Broker 配置

一旦创建了 Broker 配置，接着需要启用该配置才能使用它。为此使用以下命令：

```
enable configuration;
```

需要一些时间来完成此命令。该命令完成后，发出以下命令：

```
show configuration;
```

如果一切正常，将显示“SUCCESS”消息。

若非如此，则需要检查以下日志文件并解决所有问题：

```
$ORACLE_BASE/diag/rdbms/<db_name>/<instance_name>/trace/  
drcINSTANCE_NAME.log
```

另一个有用的命令为：

```
show database verbose DB_NAME
```

注意：该文件出现在所有节点上。

注意：用 DB_UNIQUE_NAME 的值替代 <db_name>

注意：用 ORACLE_SID 的值替代 <instance_name>。

注意：在启用配置前不显示辅助实例。

4.5.6 Broker 自定义

这将提供基本配置。备用数据库一旦启动，Data Guard Broker 将按上面所述自动将备用数据库置于托管恢复模式。

但是，该基本配置不足以维持整个环境。因此必须进一步进行自定义，这可以通过设置 Data Guard 属性来实现。应定义以下属性：

```
dgmgrl sys/mypasswd
```

```
edit database MOON set property LogArchiveMaxProcesses=5;
```

```
edit database SUN set property LogArchiveMaxProcesses=5;
```

```
edit database MOON set property StandbyFileManagement=auto;
```

```
edit database SUN set property StandbyFileManagement=auto;
```

4.5.6.1 最高可用性/最大保护

此外，如果使用最高可用性/最大保护模式，还需要设置以下参数值：

```
edit database MOON set property LogXptMode='SYNC';
```

```
edit database SUN set property LogXptMode='SYNC';
```

```
edit configuration
```

```
protection mode as maxavailability | maxprotection;
```

set

5 监视

5.1.1 引言

创建了配置之后，需要检查一切是否顺利运行。下面几节介绍监视环境的不同方法。

5.1.2 日志文件

归档操作发生时，该操作会被记入警报日志。每当主数据库上发生日志切换，都将记录到警报日志中。当在备用数据库上执行归档日志应用时，这将记录到备用实例的警报之日中。

5.1.3 固定视图

可以使用以下固定视图来监视 Data Guard Broker:

主站点	
V\$ARCHIVE_DEST	说明当前实例的所有归档重做日志目标，及其当前值、模式和状态。
V\$ARCHIVE_DEST_STATUS	显示归档重做日志目标的运行时信息和配置信息。
V\$ARCHIVED_LOG	显示来自控制文件的归档重做日志信息，包括归档日志名称。
V\$DATABASE	提供来自控制文件的数据库信息。包括数据库的状态。
V\$LOG	该视图包含来自联机重做日志的日志文件信息。
备用站点	
V\$ARCHIVED_LOG	显示来自控制文件的归档重做日志信息，包括归档日志名称。
V\$DATABASE	提供来自控制文件的数据库信息，包括数据库的状态。
V\$LOGFILE	包含有关联机/备用重做日志的信息。
V\$MANAGED_STANDBY	显示与 Data Guard 有关的某些 Oracle 数据库服务器进程的当前信息和状态信息。
V\$STANDBY_LOG	显示有关备用重做日志的信息。

6 管理

6.1 转换

有时，由于计划中断（如在主服务器上进行维护），主站点将变为不可用，在这种情形下，临时转换到备用站点是十分有益的。这一过程是非破坏性的，并且在主站点恢复可用后可以执行反向转换，整个过程无需重建任何一个数据库。

6.1.1 使用 SQL Plus 执行转换

验证每个数据库都针对其将要承担的角色进行了合适的配置，并且备用数据库处于挂载状态。

在从 RAC 主数据库执行转换前，除了一个主实例之外关闭所有其他实例（转换完成后可以重启这些实例）。

在执行向 RAC 备用数据库的转换或故障切换前，除一个备用实例之外关闭所有其他实例（角色转换完成后可以重启这些实例）。

以下 SQL 语句执行向备用数据库的转换（注意：如果需要恢复，请按 7.1 节中的步骤应用重做日志变更。

在主数据库上，发起转换：

```
alter database commit to switchover to physical standby  
[with session shutdown];
```

注意： 如果存在到主数据库的连接，请使用“with sessin shutdown”子句。重启并挂载新的备用数据库。

在（原来的）备用数据库上，转换到新的主数据库角色：`alter database commit to switchover to primary;`

重启并打开新的主数据库。

在新的备用数据库上启动重做应用：

```
alter database recover managed standby database  
[using current logfile] disconnect;
```

注意： 如果主站点会在较长时间内不可用，通常的最佳实践是，将 `log_archive_dest_state_2` 设置为“defer”。

6.1.2 使用 Data Guard Broker 执行转换

在本案例中，通过在 Broker 的命令行界面中输入下列命令完成到备用数据库的转换（也可以从 Grid Control 完成这一操作）：

```
dgmgrl sys/mypasswd
switchover to SUN;
```

**用您希望转换到的数据库的名称替代 <SUN>。

注意：如果主站点会在较长时间内不可用，则禁用该数据库：

```
disable database MOON;
```

6.2 故障切换

6.2.1 使用 SQL Plus 执行故障切换

如果主数据库因灾难而不可用，将需要执行到备用站点的故障切换。

在打开备用数据库实现正常运营前，需要将所有未完成的变更应用到该数据库。为促进故障切换的完成，需要完成下列步骤：

通过以下命令终止托管恢复模式：

```
alter database recover managed standby database finish;
```

将备用数据库切换为主数据库角色，并以正常模式打开它：

```
alter database commit to switchover to primary;
alter database open;
```

6.2.2 使用 Data Guard Broker 执行故障切换

使用 Data Guard Broker 同样可以完成故障切换。其命令为：

```
dgmgrl sys/mypasswd
failover to SUN;
```

6.3 强制故障切换

在某些情况下，有可能无法执行 *标准* 故障切换。如果是这样，可以执行 *强制* 故障切换。强制故障切换是破坏性的，一旦激活强制故障切换，所有剩余的备用数据库都将需要重建。

如果使用 SRL，则发出：

```
alter database recover managed standby database finish force;
```

然后执行下列步骤：

```
alter database activate standby database [skip standby logfile];
```

```
shutdown immediate;
startup mount;
alter database open resetlogs;
```

6.3.1 使用 Data Guard Broker 执行强制故障切换

使用 Data Guard Broker 同样可以完成强制故障切换:

```
dgmgrl sys/mypasswd
failover to SUN immediate;
```

6.4 以只读方式打开备用数据库

可以以只读模式打开备用数据库以运行报表或执行数据检查。通过以下方法可完成这一任务:

1. 停止托管恢复

```
alter database recover managed standby database cancel;
```

2. 打开数据库

```
alter database open read only;
```

或者, 通过在备用数据库上执行“正常”启动也会以只读方式打开它。

注意: 尽管是以只读方式打开的, 但仍会接收变更, 但是不应用到备用数据库。

6.5 实时应用/实时查询

Oracle 11g 引入了一种 Data Guard 功能, 这就是, 数据库在以只读模式打开的情况下仍可以应用重做变更。这称为实时应用/实时查询。该功能可通过 SQL Plus 以下列方式启用:

1. 连接到备用数据库:

```
sqlplus / as sysdba
```

2. 取消托管恢复

```
alter database recover managed standby database cancel;
```

3. 打开数据库

```
alter database open;
```

4. 重启托管恢复

```
alter database recover managed standby database disconnect;
```

7 附录 A – 使用 RMAN 创建备用数据库（传统方法）

在 Oracle 11g 之前，不能执行“动态”备用数据库创建。因此当时广泛采用以下方法，现在对于 11g 来说，如果愿意仍可采用这一方法。

7.1 前提

该过程可能比较复杂和灵活，不过下面的过程假设以下前提：

- 备用数据库的结构与主数据库相同。
- 已存在恢复目录。

注意：按这里描述的过程建立备用数据库不需要使用恢复目录

注意：在运行中的 Data Guard 环境中使用 RMAN 通常需要恢复目录。

7.2 RMAN 备份

有两个创建 RMAN 备份的方法，该备份适于创建备用数据库。

7.2.1 新备份

如果以前从未使用 RMAN 对数据库做过备份，则需要进行备份：

从主站点：

```
rman target /

Run {
  allocate channel d1 device type disk format '/backup/%U';
  allocate channel d2 device type disk format '/backup/%U';

  sql "alter system switch logfile"
  backup database include current controlfile for standby plus
  archivelog;
}
```

在 RAC 实现中发出：

```
Run {
  allocate channel d1 device type disk format '/backup/%U';
  allocate channel d2 device type disk format '/backup/%U';

  sql "alter system archive log current"
  backup database include current controlfile for standby;
  sql "alter system archive log current";
  backup archivelog all;
}
```

7.2.2 现有备份

如果之前已对数据库做过备份，则需要创建备用控制文件，为此，执行以下命令：

```
rman target /
run {
    allocate channel d1 device type disk format '/backup/%U';
    allocate channel d2 device type disk format '/backup/%U';

    backup current controlfile for standby;
    sql 'alter system archive log current;';
    backup archive log all not backed up 1 times;
}
```

7.3 创建备用数据库

7.3.1 先决条件

在构建备用数据库之前，请确保：

- 备用计算机上已经安装了 Oracle 可执行文件
- 备用计算机上已存在数据库目录结构
- 备用计算机上启动了监听器
- 备用数据库已静态注册到监听器
- 主数据库上存在 `tnsnames.ora` 条目，该条目指向备用数据库
- 备用主机上的主初始化参数文件可用
- `remote_password_file` 设置为 `none`
- 备用实例已经创建了外部口令文件
- RMAN 备份集可用于备用主机

7.3.2 过程

1. 启动备用数据库实例：

```
startup nomount pfile=init.ora
```

2. 连接到 RMAN（在主站点上）

```
rman target / auxiliary sys/syspwd@<standby_alias>
```

3. 创建备用数据库：

```
duplicate target database for standby dorecover
[nofilenamecheck];
```

现在将创建备用数据库。

注意：在将备用数据库投入运营前，不一定要恢复它，因为在托管恢复启动的情况下，将通过重做日志差异解析功能同步备用数据库。因此可能不使用“dorecover”选项。

8 附录 B – 其他阅读材料

8.1 Oracle 手册

- **Oracle Clusterware 安装指南**
适用于 Linux 的 11g 第 1 版
Oracle Corporation 编号 B28263-03 2007 年 10 月
- **Oracle Real Application Clusters 安装指南**
适用于 Linux 的 11g 第 1 版
Oracle Corporation 编号 B28264-02 2007 年 9 月
- **Oracle Data Guard 概念和管理**
11g 第 1 版
Oracle Corporation 编号 B28294-02 2007 年 9 月
- **Oracle Data Guard Broker**
11g 第 1 版
Oracle Corporation 编号 B28295-02 2007 年 9 月
- **Oracle 数据库备份和恢复用户指南**
11g 第 1 版
Oracle Corporation 编号 B28270-02 2007 年 9 月
- **Oracle 数据库备份和恢复参考**
11g 第 1 版
Oracle Corporation 编号 B28273-02 2007 年 9 月

8.2 Metalink

- 说明 413484.1
Data Guard 支持同一个 Data Guard 配置中的异构主系统和备用系统

