Oracle 11g DataGuard配置过程-CentOS环境

## 目的

本文描述了在CentOS 6.5环境下，如果安装与配置Oracle 11g的DataGuard，实现在线日志的主库与备库物理同步传输（而不是异步传输）。 本实验采用RMAN+DUPLICATE命令方式将主库数据库文件、日志文件、控制文件共同复制到备库中，不需要主库关机，也不需要主库做备份，再到备库恢复，方便快捷，但需要较快的网络环境。

本次配置的数据保护模式为“maximize performance”（最大性能模式），它是默认的GataGuard体系中的数据保护模式。在最大性能模式下，redo传输可以选择arch或者lgwr进程，主库的日志可以异步传输到备库中，对于主库性能影响自然是最小的，不过由于redo传输不是即时，主库崩溃时会出现数据丢失。

## 参考网站

* oracle11g dataguard 完全手册:

<https://blog.csdn.net/cuiyan1982/article/details/52333697>

* [使用RMAN备份集搭建Oracle Dataguard Step by Step（一）](http://blog.itpub.net/17203031/viewspace-1170268/)（二）（三）:

<http://blog.itpub.net/17203031/viewspace-1170268/>

## 重要进程

RFS (remote file server process) :接受由primary数据库的lgwr或arch通过oracle net传来的redo 数据，写入standby redo logs或standby archived redo logs

MRP(managed Recovery Process): 管理恢复进程。

FAL(fetch archive log): 在primary和standby数据库的两端同时配置。Primary端是fetch archive log server,standby端是fetch archive log client. FAL是自动探测primary/standby数据库之间archived logs是否有间隔的一个进程。

## 安装前准备

### 主库主机

* 主机名称：pc-master，**必须**安装了Oracle 11g的软件和数据库，保护模式为MAXIMUM PERFORMANCE。
* 主库的实例名称是ORCL, DB\_UNIQUE\_NAME也是ORCL。
* Hosts文件：

Hosts文件名称是/etc/hosts，定义了IP地址和新的主机名称的映射关系，目的是用名称代替IP地址，方便管理。最后两行的内容如下：

$ cat /etc/hosts

127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4

::1 localhost localhost.localdomain localhost6 localhost6.localdomain6

192.168.31.11 pc-master

192.168.31.10 pc-slave

* 环境变量：

ORACLE\_HOME=/home/oracle/app/oracle/product/11.2.0/dbhome\_1

ORACLE\_SID=ORCL

$ echo $ORACLE\_HOME

/home/oracle/app/oracle/product/11.2.0/dbhome\_1

$ echo $ORACLE\_SID

orcl

### 备库主机

* 主机名称：pc-slave，安装了Oracle 11g软件，**必须没有安装数据库**，如果已经安装了数据库，可以使用dbca命令先删除。
* 保护模式为MAXIMUM PERFORMANCE。
* 备库的实例名称是ORCL, DB\_UNIQUE\_NAME是ORCLDG。

注意，主库主机和备库主机的DB\_UNIQUE\_NAME不能相同！

* Hosts文件

同主库主机

* 环境变量

同备库主机

### 运行日志告警

在安装配置过程，需要随时查看错误日志告警，

告警日志位置查看命令: show parameter background\_dump\_dest;

参见：<http://blog.chinaunix.net/uid-17069315-id-5736359.html>

在主库中是：tail -f /home/oracle/app/oracle/diag/rdbms/**orcl**/orcl/trace/alert\_orcl.log

在备库中是: tail -f /home/oracle/app/oracle/diag/rdbms/**orcldg**/orcl/trace/alert\_orcl.log

## 主库配置

### listener.ora

文件名：C:\app\Administrator\oracle\product\11.2.0\dbhome\_1\network\admin\listener.ora

# listener.ora Network Configuration File: C:\app\Administrator\oracle\product\11.2.0\dbhome\_1\network\admin\listener.ora

# Generated by Oracle configuration tools.

#修改SID\_LIST\_LISTENER

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(SID\_NAME = CLRExtProc)

(ORACLE\_HOME = C:\app\Administrator\oracle\product\11.2.0\dbhome\_1)

(PROGRAM = extproc)

(ENVS = "EXTPROC\_DLLS=ONLY:C:\app\Administrator\oracle\product\11.2.0\dbhome\_1\bin\oraclr11.dll")

)

#添加SID\_DESC

(SID\_DESC =

(GLOBAL\_DBNAME = orcl)

(ORACLE\_HOME = C:\app\Administrator\oracle\product\11.2.0\dbhome\_1)

(SID\_NAME = orcl)

)

)

#修改LISTENER

LISTENER =

(DESCRIPTION\_LIST =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))

#在原来的基础上添加SID\_DESC

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

)

)

ADR\_BASE\_LISTENER = C:\app\Administrator\oracle

### tnsnames.ora

文件名：C:\app\Administrator\oracle\product\11.2.0\dbhome\_1\network\admin\ tnsnames.ora

# tnsnames.ora Network Configuration File: C:\app\Administrator\oracle\product\11.2.0\dbhome\_1\network\admin\tnsnames.ora

# Generated by Oracle configuration tools.

ORCL =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = pc-master)(PORT = 1521))

)

(CONNECT\_DATA =

(SERVICE\_NAME = orcl)

)

)

ORCLDG =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = pc-slave)(PORT = 1521))

)

(CONNECT\_DATA =

(SERVICE\_NAME = orcldg)

)

)

EXTPROC\_CONNECTION\_DATA =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1))

)

(CONNECT\_DATA =

(SID = PLSExtProc)

(PRESENTATION = RO)

)

)

ORACLR\_CONNECTION\_DATA =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))

)

(CONNECT\_DATA =

(SID = CLRExtProc)

(PRESENTATION = RO)

)

)

### 参数文件

参数文件是配置DataGuard的核心文件，有文本类型(initorcl.ora)和填制系统类型(spfileorcl.ora)两种。这两个文件都在目录/home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/中。

文本类型(initorcl.ora)文件名：/home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/initorcl.ora

orcl.\_\_db\_cache\_size=251658240

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

orcl.\_\_large\_pool\_size=16777216

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

orcl.\_\_pga\_aggregate\_target=536870912

orcl.\_\_sga\_target=603979776

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

orcl.\_\_shared\_pool\_size=301989888

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

\*.audit\_file\_dest='/home/oracle/app/oracle/admin/orcl/adump'

\*.audit\_trail='db'

\*.compatible='11.2.0.0.0'

\*.control\_files='/home/oracle/app/oracle/oradata/orcl/control01.ctl','/home/oracle/app/oracle/flash\_recovery\_area/orcl/control02.ctl'

\*.db\_block\_size=8192

\*.db\_domain=''

\*.db\_file\_name\_convert='orcl','orcl'

\*.db\_name='orcl'

\*.db\_recovery\_file\_dest='/home/oracle/app/oracle/flash\_recovery\_area'

\*.db\_recovery\_file\_dest\_size=4070572032

\*.db\_unique\_name='ORCL'

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

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

\*.fal\_client='orcl'

\*.fal\_server='orcldg'

\*.log\_archive\_config='dg\_config=(orcl,orcldg)'

\*.log\_archive\_dest\_1='LOCATION=/home/oracle/app/oracle/flash\_recovery\_area/ORCL/archivelog/ valid\_for=(all\_logfiles,all\_roles) db\_unique\_name=ORCL'

\*.log\_archive\_dest\_2='SERVICE=orcldg lgwr sync valid\_for=(online\_logfiles,primary\_role) db\_unique\_name=orcldg'

\*.log\_archive\_dest\_state\_1='ENABLE'

\*.log\_archive\_dest\_state\_2='ENABLE'

\*.log\_archive\_max\_processes=10

\*.log\_file\_name\_convert='/home/oracle/app/oracle/oradata/orcl/','/home/oracle/app/oracle/oradata/orcl/'

\*.memory\_target=1132462080

\*.open\_cursors=300

\*.processes=150

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

\*.standby\_file\_management='AUTO'

\*.undo\_tablespace='UNDOTBS1'

其中，参数fal\_client过时了，没有意义。

该参数文件是PFILE，不是SPFILE，修改该文件后，不会立即起作用，必须这样才能起作用：

SQL>shutdown immediate;

SQL>create spfile from pfile;

SQL>startup;

### 密码文件

文件名：C:\app\Administrator\oracle\product\11.2.0\dbhome\_1\database\PWDorcl.ora

该文件必须复制到备库的相同目录中。

### 创建standby类型在线日志文件

对于主库，standby类型的日志文件是不需要存在的。但为了以后主库转为备库，还是需要增加。

要手工增加standby类型的logfile，每个logfile的大小应该大于等于普通的logfile文件大小。Standby logfile的数量应该大于普通日志文件的数量：

在主库是打开open状态的情况下：

alter database add standby logfile group 11 '/home/oracle/app/oracle/oradata/orcl/standby11.log' size 500M;

alter database add standby logfile group 12 '/home/oracle/app/oracle/oradata/orcl/standby12.log' size 500M;

alter database add standby logfile group 13 '/home/oracle/app/oracle/oradata/orcl/standby13.log' size 500M;

alter database add standby logfile group 14 '/home/oracle/app/oracle/oradata/orcl/standby14.log' size 500M;

可以随时删除standby logfile,命令是(主库是mount状态)：

alter database drop standby logfile group 11;

alter database drop standby logfile group 12;

alter database drop standby logfile group 13;

alter database drop standby logfile group 14;

host rm /home/oracle/app/oracle/oradata/orcl/standby11.log

host /home/oracle/app/oracle/oradata/orcl/standby12.log

host /home/oracle/app/oracle/oradata/orcl/standby13.log

host /home/oracle/app/oracle/oradata/orcl/standby14.log

可以通过以下命令查询所有在线日志文件：

SQL> select \* from v$logfile;

SQL> select group#, dbid from v$standby\_log;

## 备库配置与创建

注意，这时备库只安装了软件，没有安装数据库。

### 从主库复制网络文件、参数文件和密码文件。

需要将主库的参数文件和密码文件复制到备库中。在主库运行以下命令：

#下面复制网络配置文件

$ cd /home/oracle/app/oracle/product/11.2.0/dbhome\_1/network/admin

$ scp \* pc-slave:/home/oracle/app/oracle/product/11.2.0/dbhome\_1/network/admin

#下面开始复制参数文件和密码文件

$ cd /home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/

$ scp initorcl.ora pc-slave:/home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/

$ scp orapworcl pc-slave:/home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/

### 修改备库的listener.ora

文件名：/home/oracle/app/oracle/product/11.2.0/dbhome\_1/network/admin \listener.ora

# listener.ora Network Configuration File: /home/oracle/app/oracle/product/11.2.0/dbhome\_1/network/admin/listener.ora

# Generated by Oracle configuration tools.

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(SID\_NAME = PLSExtProc)

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

(PROGRAM = extproc)

)

(SID\_DESC =

(GLOBAL\_DBNAME = orcldg)

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

(SID\_NAME = orcl)

)

)

LISTENER =

(DESCRIPTION\_LIST =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))

(ADDRESS = (PROTOCOL = TCP)(HOST = pc-slave)(PORT = 1521))

)

)

ADR\_BASE\_LISTENER = /home/oracle/app/oracle

INBOUND\_CONNECT\_TIMEOUT\_listener=10

listener.ora修改完成后，启动监听器：

$lsnrctl start

结果有包括下面的服务，其中的orcldg就是listener.ora中的静态监听名称GLOBAL\_DBNAME。

Service "orcldg" has 2 instance(s).

Instance "orcl", status UNKNOWN, has 1 handler(s) for this service...

### 不用修改tnsnames.ora和密码文件orapworcl

tnsnames.ora以及orapworcl与主库完全相同，不能修改。

### 修改备库的参数文件

参数文件是配置DataGuard的核心文件，有文本类型(initorcl.ora)和填制系统类型(spfileorcl.ora)两种。这两个文件都在目录/home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/中。

文本类型(initorcl.ora)文件名：/home/oracle/app/oracle/product/11.2.0/dbhome\_1/dbs/initorcl.ora

orcl.\_\_db\_cache\_size=251658240

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

orcl.\_\_large\_pool\_size=16777216

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

orcl.\_\_pga\_aggregate\_target=536870912

orcl.\_\_sga\_target=603979776

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

orcl.\_\_shared\_pool\_size=301989888

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

\*.audit\_file\_dest='/home/oracle/app/oracle/admin/orcl/adump'

\*.audit\_trail='db'

\*.compatible='11.2.0.0.0'

\*.control\_files='/home/oracle/app/oracle/oradata/orcl/control01.ctl','/home/oracle/app/oracle/flash\_recovery\_area/orcl/control02.ctl'

\*.db\_block\_size=8192

\*.db\_domain=''

\*.db\_file\_name\_convert='orcl','orcl'

\*.db\_name='orcl'

\*.db\_recovery\_file\_dest='/home/oracle/app/oracle/flash\_recovery\_area'

\*.db\_recovery\_file\_dest\_size=4070572032

\*.db\_unique\_name='ORCLDG'

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

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

\*.fal\_client='orcldg'

\*.fal\_server='orcl'

\*.log\_archive\_config='dg\_config=(orcl,orcldg)'

\*.log\_archive\_dest\_1='LOCATION=/home/oracle/app/oracle/flash\_recovery\_area/ORCL/archivelog/ valid\_for=(all\_logfiles,all\_roles) db\_unique\_name=ORCLDG'

#下面的lgwr sync表示在线日志的同步传输。

\*.log\_archive\_dest\_2='SERVICE=orcl lgwr sync valid\_for=(online\_logfiles,primary\_role) db\_unique\_name=orcl'

\*.log\_archive\_dest\_state\_1='ENABLE'

\*.log\_archive\_dest\_state\_2='ENABLE'

\*.log\_archive\_max\_processes=10

\*.log\_file\_name\_convert='/home/oracle/app/oracle/oradata/orcl/','/home/oracle/app/oracle/oradata/orcl/'

\*.memory\_target=1132462080

\*.open\_cursors=300

\*.processes=150

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

\*.standby\_file\_management='AUTO'

\*.undo\_tablespace='UNDOTBS1'

其中，参数fal\_client过时了，没有意义。

接下来必须保证在备库中，initorcl.ora中的所有提到的目录在备库中都存在,如果不存在，就要手工创建这些空目录。而且必须保证除了上述文件之外，备库中的其他文件都要全部删除（数据文件，归档日志文件，在线重做日志文件，控制文件），然后将系统启动到nomount，以便从主库复制数据到备库中。

$sqlplus / as sysdba

SQL>shutdown immediate;

#下面的操作生成进制的系统参数文件参数spfileorcl.ora：即initorcl.ora生产spfileorcl.ora

#只有这样参数文件才能起作用

SQL>**create spfile from pfile;**

SQL>startup nomount;

startup nomount命令必须成功，否则无法从主库复制数据文件，控制文件等其他文件。

### 从主库在线克隆到备库

到目前为止，以上配置工作都全部完成，主库是正常的open状态，备库是nomount状态。现在可以开始复制主库了。

以下命令在主库或者备库上均可运行：

$rman target sys/123@orcl auxiliary sys/123@orcldg

rman>duplicate target database for standby from active database dorecover nofilenamecheck;

其中dorecover nofilenamecheck必须要加上，这是因为主库和备库结构相同，不能作文件检查。参见：<http://blog.51cto.com/gavinshaw/1592737>

经过这个步骤，如果没有任何错误，主库的数据文件，归档日志文件，控制文件，standby类型控制文件，在线重做日志文件（含主库新增的standby类型的在线重做日志文件）等一切所需要的文件都复制到备库中来了。相当于将主库克隆到备库了。

克隆顺利完成后，备库就可以打开了，以下将备库打开为只读模式。

C:\>sqlplus / as sysdba

SQL>shutdown immediate

SQL>startup nomount

SQL>alter database mount standby database;

#以只读方式打开备库

SQL> alter database open read only;

### 创建standby类型在线日志文件

如果在上一步在线复制中，主库已经添加了standby类型的在线日志文件，这些文件会自动克隆到备库中，这一步可以省略了。如果主库有没有添加standby类型的在线日志文件，则需要手工在备库中添加。

对于备库，如果要做在线日志的主备同步，则是必须存在的。创建方式同主库。

注意，创建standby类型的日志文件的前提是备库先正常open。必须先在备库端进行standby类型控制文件和数据文件的恢复之后，打开了数据库之后，才能创建standby日志文件。

## 启动

### 备库先启动

* 首次启动为备用库模式，以下命令在备库中运行：

C:\>lsnrctl start

C:\>sqlplus / as sysdba

SQL>shutdown immediate

SQL>startup nomount

SQL>alter database mount standby database;

#可选

SQL>alter database activate standby database;

#以只读方式打开备库

SQL> alter database open read only;

**#开启后台同步主库在线日志的MRP进程：**

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

SQL>shutdown immediate

如果要停止同步，运行这样的命令：

SQL>alter database recover managed standby database cancel;

如果只想同步归档日志，不同步在线日志，可以运行这样的命令：

SQL>alter database recover managed standby database disconnect from session;

* 以后常态化启动，以下命令在备库中运行：

C:\>lsnrctl start

C:\>sqlplus / as sysdba

SQL>startup

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

### 主库后启动

C:\>lsnrctl start

C:\>sqlplus / as sysdba

SQL>startup

## 数据同步测试

### 主库运行

* 归档日志同步：切换日志，将归档日志同步到备库中。以下命令在主库中运行：

alter system switch logfile;

* 在线日志同步：创建表，维护表，注意数据修改之后要执行commit才能同步数据到备库中。

SQL>insert into c values(10);

SQL>commit;

### 备库查询

* 在备库查看表中的数据是否从主库同步过来了。

SQL>select \* from c;

### 日志序列查询

* 以下命令在主库和备库分别运行

SQL> archive log list;

如果结果中的当前日志序列值是相同的，表示同步正确，否则是不正确。

## 主备切换

Oracle的主备切换非常简单，只要主库和备库都工作正常，不需要在主库或者备库中修改任何参数文件。

* 主库切换为备库

确保原主库端运行正常，在原主库端open的情况下，在原主库端运行以下命令：

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

数据库已更改。

SQL> shutdown immediate

SQL> startup nomount

SQL> alter database mount standby database;

数据库已更改。

SQL> alter database open read only;

数据库已更改。

这时原主库已更改为备库，而且是只读模式，但还没有执行同步命令，因为这时原备库还没有转为主库。

* 备库切换为主库

确保原备库端运行正常，在原备库端open的情况下，先关闭物理备库的当前连接回话，再重新打开，在原备库端运行以下命令：

SQL> alter database commit to switchover to primary with session shutdown;

数据库已更改。

SQL> alter database open;

数据库已更改。

这时原备库已经切换为主库了。

* 新备库开始在后台同步新主库的数据

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

现在可以做主备转换之后的数据同步测试了。

## 主库损坏，备库变为主库

主库如果损坏，完全无法工作，这时备库可以变为主库，继续工作。先关闭物理备库的当前连接回话，再重新打开，在原备库端运行以下命令：

SQL> Alter database commit to switchover to physical standby with session shutdown;

Database altered.

SQL> alter database open;

SQL> alter database recover managed standby database finish;

Database altered.

#下一步可选

SQL>alter database recover managed standby database cancel;

#查看switchover\_status，仍然是TO PRIMARY

SQL> select switchover\_status from v$database;

SWITCHOVER\_STATUS

--------------------

TO PRIMARY

**#切换到主库**

SQL> alter database commit to switchover to primary;

Database altered.

#再查看switchover\_status，为NOT ALLOWED，正常了

SQL> select switchover\_status from v$database;

SWITCHOVER\_STATUS

--------------------

NOT ALLOWED

SQL> select database\_role from v$database;

DATABASE\_ROLE

----------------

PRIMARY

## 常用查询

* 查看打开类型和控制文件类型，查询是主机还是备机。

select open\_mode,controlfile\_type from v$database;

主库查询结果：open\_mode=**READ WRITE** controlfile\_type=**CURRENT**

备库查询结果：open\_mode=**READ ONLY** controlfile\_type=**STANDBY**

* 查看switchover\_status

select switchover\_status from v$database;

在主库端正常的结果是：TO STANDBY

在备库端正常的结果是：NOT ALLOWED

* 查看在线重做日志文件（含standby类型的日志文件）

select \* from v$logfile;

* 查看日志目标设置是否正确，status都是VALID才是正确的

select error,status,process from v$archive\_dest where rownum<3;

* 查看保护模式

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

* 查看在线日志

select group#, sequence#, archived, status from v$log;