

**期末项目设计报告**

|  |  |  |  |
| --- | --- | --- | --- |
| 题 目 | 基于Oracle的聊天及管理的数据库设计 | | |
| 课程 | Oracle数据库应用 | | |
| 学 院 | 信息科学与工程学院 | | |
| 专 业 | 软件工程 | 年级 | 2017级 |
| 学生姓名 | 罗川龙 | 学号 | 201710414311 |
| 指导教师 | 赵卫东 | 职称 | 副教授 |

|  |  |  |  |
| --- | --- | --- | --- |
| **评分项** | **评分标准** | **满分** | **得分** |
| 文档整体 | 文档内容详实、规范，美观大方 | 10 |  |
| 表设计 | 表，表空间设计合理，数据合理 | 20 |  |
| 用户管理 | 权限及用户分配方案设计正确 | 10 |  |
| PL/SQL设计 | 存储过程和函数设计正确 | 25 |  |
| 备份方案 | 备份方案设计正确 | 25 |  |
| 容灾方案 | DataGuard设计正确 | 10 |  |
| **得分合计** | | |  |

2019 年 11 月 18 日

1.引言

1.1目的

随着互联网的迅速普及，以及网络技术的不断发展，人们通过网络的交流方式不断发展。网络聊天室就是其中的一种。聊天室的系统即使交流的方式满足了网络中多人同时交流的需要。

本设计所完成的聊天室具有常用聊天室的所有功能。包括多人在线的同时聊天，两人间的私密聊天。在聊天的时候可以发送表情，还能显示在线用户数量，满足了不同人群的交流方式。

1.2背景

Internet是目前世界上最大的计算机互联网络，它遍布全球，将世界各地各种规模的网络连接成一个整体。作为Internet上一种先进的，易于被人们所接受的信息检索手段，World Wide Web(简称WWW)发展十分迅速，成为目前世界上最大的信息资源宝库。据估计，目前Internet上已有上百万个Web站点，其内容范围跨越了教育科研、文化事业、金融、商业、新闻出版、娱乐、体育等各个领域，其用户群十分庞大，因此，建设一个好的Web站点对于一个机构的发展十分重要。近年来计算机技术的快速发展，特别是计算机网络的发展，越来越深刻的改变了人们生活的方方面面。使得人们能以更低廉的价格，开发出更方便、更实用的网络工具。各种在线服务系统，更是深刻的影响了人们的联系和交流方式，使得人们可以在远隔千里之遥随时通讯。过去的种种陈旧的联系方式，已经不能满足现代生活的需要。网上聊天系统作为一种方便消费者与人们之间联系的实用系统便应运而生。网上聊天系统是为人们进行交流和联系提供的一个平台。通过提供完善的网上聊天系统的管理，可以达到增进人们与消费者之间、消费者与消费者之间的交流和联系的目的。

1.3定义

“聊天室(chat room)是一个网上空间，为了保证谈话的焦点，聊天室通常有一定的谈话主题。任何一个联入Internet、使用正确的聊天软件，并且渴望谈论的人都可以享受其乐趣。聊天室会话是自然会话在信息时代的延伸。合作原则解释不了聊天室会话的诸多问题,没有其适用性。聊天室有语音聊天室和视频聊天室等分类。”

2.环境搭建

# 2.1技术支持

2.11

Linux是一套免费使用和自由传播的类Unix操作系统，是一个基于POSIX和Unix的多用户、多任务、支持多线程和多CPU的操作系统。伴随着互联网的发展，Linux得到了来自全世界软件爱好者、组织、公司的支持。它除了在服务器操作系统方面保持着强劲的发展势头以外，在个人电脑、嵌入式系统上都有着长足的进步。使用者不仅可以直观地获取该操作系统的实现机制，而且可以根据自身的需要来修改完善这个操作系统，使其最大化地适应用户的需要。  

Linux不仅系统性能稳定，而且是开源软件。其核心防火墙组件性能高效、配置简单，保证了系统的安全。在很多企业网络中，为了追求速度和安全，Linux操作系统不仅仅是被网络运维人员当作服务器使用，Linux既可以当作服务前，又可以当作网络防火墙是Linux的 一大亮点。

Linux与其他操作系统相比 ，具有开放源码、没有版权、技术社区用户多等特点 ，开放源码使得用户可以自由裁剪，灵活性高，功能强大，成本低。尤其系统中内嵌网络协议栈 ，经过适当的配置就可实现路由器的功能。这些特点使得Linux成为开发路由交换设备的理想开发平台。

2.12

ORACLE数据库系统是美国ORACLE公司（甲骨文）提供的以分布式数据库为核心的一组软件产品，是目前最流行的客户/服务器(CLIENT/SERVER)或B/S体系结构的数据库之一。比如SilverStream就是基于数据库的一种中间件。ORACLE数据库是目前世界上使用最为广泛的数据库管理系统，作为一个通用的数据库系统，它具有完整的数据管理功能；作为一个关系数据库，它是一个完备关系的产品；作为分布式数据库它实现了分布式处理功能。但它的所有知识，只要在一种机型上学习了ORACLE知识，便能在各种类型的机器上使用它。

Oracle数据库最新版本为Oracle Database 12c。Oracle数据库12c 引入了一个新的多承租方架构，使用该架构可轻松部署和管理数据库云。此外，一些创新特性可最大限度地提高资源使用率和灵活性，如Oracle Multitenant可快速整合多个数据库，而Automatic Data Optimization和Heat Map能以更高的密度压缩数据和对数据分层。这些独一无二的技术进步再加上在可用性、安全性和大数据支持方面的主要增强，使得Oracle数据库12c 成为私有云和公有云部署的理想平台。

2.13

VMware Workstation（中文名“威睿工作站”）是一款功能强大的桌面[虚拟计算机](https://baike.baidu.com/item/%E8%99%9A%E6%8B%9F%E8%AE%A1%E7%AE%97%E6%9C%BA/5732003" \t "https://baike.baidu.com/item/_blank)软件，提供用户可在单一的桌面上同时运行不同的操作系统，和进行开发、测试 、部署新的应用程序的最佳解决方案。VMware Workstation可在一部实体机器上模拟完整的网络环境，以及可便于携带的[虚拟机](https://baike.baidu.com/item/%E8%99%9A%E6%8B%9F%E6%9C%BA" \t "https://baike.baidu.com/item/_blank)器，其更好的灵活性与先进的技术胜过了市面上其他的虚拟计算机软件。对于企业的 IT开发人员和系统管理员而言， VMware在虚拟网路，实时快照，拖曳[共享文件夹](https://baike.baidu.com/item/%E5%85%B1%E4%BA%AB%E6%96%87%E4%BB%B6%E5%A4%B9/7258865" \t "https://baike.baidu.com/item/_blank)，支持 PXE 等方面的特点使它成为必不可少的工具。

2.14

[Microsoft](https://baike.baidu.com/item/Microsoft" \t "https://baike.baidu.com/item/_blank)Windows操作系统是美国[微软公司](https://baike.baidu.com/item/%E5%BE%AE%E8%BD%AF%E5%85%AC%E5%8F%B8" \t "https://baike.baidu.com/item/_blank)研发的一套操作系统，它问世于1985年，起初仅仅是Microsoft-DOS模拟环境，后续的系统版本由于微软不断的更新升级，不但易用，也当前应用最广泛的操作系统。 [1]

Windows采用了图形化模式GUI，比起从前的Dos需要输入指令使用的方式，更为人性化。随着计算机硬件和软件的不断升级，微软的 Windows也在不断升级，从架构的16位、32位再到64位,系统版本从最初的 Windows1.0到大家熟知的 Windows95、 Windows98、 [Windows2000](https://baike.baidu.com/item/Windows2000/2174641" \t "https://baike.baidu.com/item/_blank)、 [Windows XP](https://baike.baidu.com/item/Windows XP/191927" \t "https://baike.baidu.com/item/_blank)、 Windows Vista、 [Windows7](https://baike.baidu.com/item/Windows7/1238780" \t "https://baike.baidu.com/item/_blank)、Windows8、Windows8.1、Windows10和 Windows Server服务器企业级操作系统，不断持续更新，微软一直在致力于Windows操作系统的开发和完善。

2.2表的设计

用户表t\_user

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名 |  | 数据类型 | 为空 | |  | 注释 | |  |
| Userid | | NUMBER(20) |  | N | |  | 用户编号，主键 | |  |
| Pwd | | VARCHA2(100) |  | N | |  | 密码 | |  |
| Nickname | | VARCHA2(10) |  | N | |  | 昵称 | |  |
| Sex | | NUMBER(1) |  | N | |  | 性别，0或1 | |  |
| Birthday | | DATE |  | Y | |  | 生日 | |  |
| Currstate | | NUMBER（1） |  | N | |  | 当前状态 | |  |
| Policy | | NUMBER(1) |  | N | |  | 交友策略，外键 | |  |

create or replace table t\_user(

 Userid NUMBER(20) primary key not null,

Pwd NVARCHAR2(100) not null,

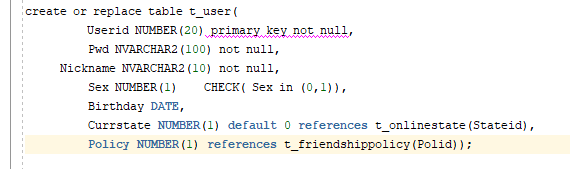
  Nickname NVARCHAR2(10) not null,

Sex NUMBER(1)   CHECK( Sex in (0,1)),

Birthday DATE,

Currstate NUMBER(1) default 0 references t\_onlinestate(Stateid),

Policy NUMBER(1) references t\_friendshippolicy(Polid));



X2U0J(35_S)A_J3EL%5D]KQ

状态表t\_onlinestate

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名 |  | 数据类型 | | 为空 |  | 注释 |
| Stateid | | NUMBER(1) |  | N |  | 状态编号，主键 |
| Statedesc | | VARCHAR2(10) |  | N |  | 状态描述，唯一 |

create table t\_onlinestate(

Stateid NUMBER(1) primary key not null，

Statedesc NVARCHAR2(10) unique not null，);

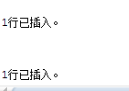
insert into t\_onlinestate values (1,’我在线上’);

insert into t\_onlinestate values (2,’离开’);

insert into t\_onlinestate values (3,’隐身’);

insert into t\_onlinestate values (4,’离线’);





好友表 t\_friend

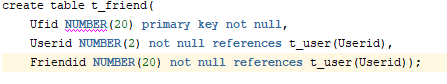
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名 |  | 数据类型 | | 为空 |  | 注释 |  |
| Ufid | | NUMBER(20) |  | N |  | 编号，主键 |  |
| Userid | | NUMBER(20) |  | N |  | 用户编号，外键 |  |
| Fridenid | | NUMBER(20) |  | N |  | 好友编号，外键 |  |

create table t\_friend(

Ufid NUMBER(20) primary key not null,

Userid NUMBER(2) not null references t\_user(Userid),

Friendid NUMBER(20) not null references t\_user(Userid));



IMG_256

聊天消息表t\_message

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名 |  | 数据类型 | | 为空 |  | 注释 |  |
| Message | | NUMBER(20) |  | N |  | 编号，主键 |  |
| Fomid | | NUMBER(20) |  | N |  | 信息发出者外键 |  |
| Toid | | NUMBER(20) |  | N |  | 信息接受者外键 |  |
| Content | | VARCHAR2(255) |  | Y |  | 信息内容 |  |
| State | | NUMBER(1) |  | N |  | 是否已读 |  |
| Sendtime | | DATE |  | N |  | 发出时间 |  |

create table t\_message(

Message NUMBER(20) primary key not null，

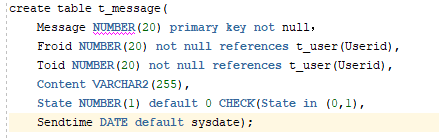
Froid NUMBER(20) not null references t\_user(Userid),

Toid NUMBER(20) not null references t\_user(Userid),

Content VARCHAR2(255),

State NUMBER(1) default 0 CHECK(State in (0,1),

Sendtime DATE default sysdate);



IMG_256

表格 5 交友策略表t\_friendshippolicy

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段名 |  | 数据类型 | | 为空 |  | 注释 |  |
| Polid | | NUMBER(1) |  | N |  | 编号，主键 |  |
| Policy | | VARCHAR2(20) |  | N |  | 交友策略，唯一 |  |

create table t\_friendshippolicy（

Polid NUMBER(1) primary key not null,

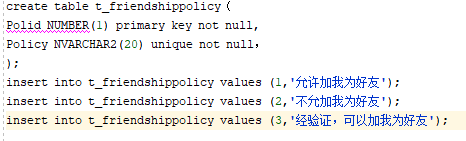
Policy NVARCHAR2(20) unique not null，

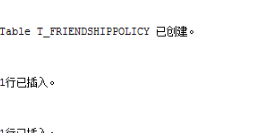
);

insert into t\_friendshippolicy values (1,'允许加我为好友');

insert into t\_friendshippolicy values (2,'不允加我为好友');

insert into t\_friendshippolicy values (3,'经验证，可以加我为好友');





存储空间分配

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 表 |  | 表空间USERS | 表空间USER01 | | | | |  |  |
| t\_user | 存储全部数据 | |  |  |  |  |
| t\_onliestate | 存储全部数据 | |  |  |  |  |
| t\_friendshippolicy | 存储全部数据 | |  |  |  |  |
| t\_friend |  | |  | 存储全部数据 |  |  |
| t\_message |  | |  | 存储全部数据 |  |  |

插入数据：

declare

pwd varchr2(40);

cu number(1);

bir DATE;

name varchar2(20);

po number(1);

sex number(1)

begin

for i in 1..30000

loop

pwd :='123456789'||i;

cu :='1'||i;

name :='li'||i;

bir :=to\_date('2017-3-2','yyyy-mm-dd');

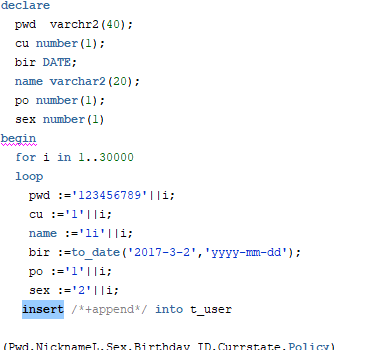
po :='1'||i;

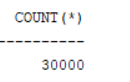
sex :='2'||i;

insert /\*+append\*/ into t\_user

(Pwd,NicknameL,Sex,Birthday\_ID,Currstate,Policy)

values (pwd,name,sex,bir,cu.po);





**3.权限及用户分配**

****创建者****：有权创建聊天室的用户。这些用户位于特定类别的“创建者”列表中：他们可以创建该类别的聊天室，还可以根据该类别分配成员身份，以及指派管理员管理聊天室。创建聊天室的用户将自动添加为聊天室的管理员。

备注

成为创建者仅仅为其提供创建聊天室的权限。通过自动升级为管理员，创建者才能够在已创建的聊天服务中进一步细化成员身份、管理员等等。

****管理者****：管理聊天室属性的用户。 聊天室管理员可以修改成员列表（添加和移除成员），以及修改聊天室管理员列表（添加和移除管理员）。 聊天室管理员可以将自己添加到成员或演示者列表（对于大会堂聊天室）中以便他们可以加入聊天室。 聊天室管理员还可以禁用聊天室（管理员可查询已禁用聊天室且可以将其永久删除）。 除聊天室的类别外，管理员可以更改聊天室的所有属性。 创建聊天室后, 只有持久聊天管理员才能更改类别。

重要

如果该管理员还是其他类别的创建者，则他/她可以更改该类别以使自己有权创建聊天室。

****成员****: 是聊天室成员的用户。 这些用户可以查看目录中的聊天室 (即使聊天室是保密的), 还可以订阅聊天室 (包括元数据选项, 如 "未读消息"、"ego 筛选器" 和 "关键字筛选器"), 并参与聊天室 (可以发布, 除非聊天室是只有演示者可以发布、获取内容和搜索的 auditorium 房间。 如果用户不是聊天室的成员, 则可以搜索聊天室 (如果他们位于该类别的 "允许的成员" 列表中), 但需要请求访问权限才能加入这些聊天室以访问内容。 (没有在系统中内置的请求访问权限或审批; 这些权限通过电子邮件、电话或其他形式的联系人外部完成。)

4.在数据库中建立一个程序包，在包中用PL/SQL语言设计一些存储过程和函数

4.1包设计：

create or replace PACKAGE MyPack1 IS

procedure register\_proc(pwd varchar2,

nickname varchar2,

sex number,

birthday date,

currstate number,

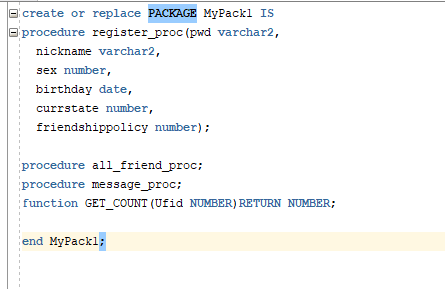
friendshippolicy number);

procedure all\_friend\_proc;

procedure message\_proc;

function GET\_COUNT(Ufid NUMBER)RETURN NUMBER;

end MyPack1;



IMG_256

## **4.2数据库存储过程的设计**

功能：将用户提供的一些基本信息输入到用户表的过程

实现：create or replace procedure register\_proc(

pwd varchar2,

nickname varchar2,

sex number,

birthday date,

currstate number,

friendshippolicy number

)

is

begin

insert into t\_user values(seq\_user.nextval,pwd,nickname,sex,birthday,currstate,friendshippolicy);

end;

功能：查询每个用户的所有好友

实现：create or replace procedure all\_friend\_proc

is

cursor all\_friend\_cur is select t\_friend.userid from t\_friend ;

v\_userid t\_friend.userid%type;

v\_friendid t\_friend.friendid%type;

v\_nickname t\_user.nickname%type;

begin

for fd in all\_friend\_cur loop

select t\_friend.userid,t\_friend.friendid,t\_user.nickname

into v\_userid,v\_friendid,v\_nickname from t\_friend,t\_user

where t\_friend.userid = t\_user.userid and t\_friend.userid = fd.userid;

dbms\_output.put\_line(v\_userid||' '||v\_friendid||' '||v\_nickname);

end loop;

end;

功能：查询每个用户和不同好友的所有聊天记录

实现：create or replace procedure message\_proc

is

v\_userid t\_user.userid%type;

v\_touserid t\_message.touserid%type;

v\_fromuserid t\_message.fromuserid%type;

v\_content t\_message.content%type;

cursor all\_fromuserid\_cur is select fromuserid from t\_message ;

begin

for fud in all\_fromuserid\_cur loop

select t\_user.userid,t\_message.touserid,t\_message.content into v\_userid,v\_touserid,v\_content from t\_message,t\_user where t\_user.userid = t\_message.fromuserid and fromuserid=fud.fromuserid;

dbms\_output.put\_line(v\_userid||' '||v\_touserid||' '||v\_fromuserid||' '||' '||v\_content);

end loop;

end;

## ****4.3数据库触发器的设计****

功能：当数据插入到T\_USER表中的时候，调用序列自增USERID

实现：create or replace trigger tir\_user

before insert on t\_user for each row

begin

select seq\_user.nextval

into :new.Userid from dual;

end;

功能：当数据插入到T\_FRIEND表中的时候，调用序列SEQ\_FRIEND自增TFID

实现：create or replace trigger tir\_fri 当数据插入到表中的时候，调用序列

before insert on T\_FRIEND for each row

begin

select seq\_friend.nextval

into :new.tfid from dual;

end;

功能：当数据插入到T\_MESSAGE表中的时候，调用序列SEQ\_MESSAGE自增MESSAGEID

实现：create or replace trigger tir\_messaged

before insert on t\_message for each row

begin

select seq\_message.nextval

into :new.MESSAGEID from dual;

end;

功能：当数据插入到T\_ADMIN表中的时候，调用序列SEQ\_ADMIN自增ADMINID

实现：create or replace trigger tir\_admin

before insert on t\_admin for each row

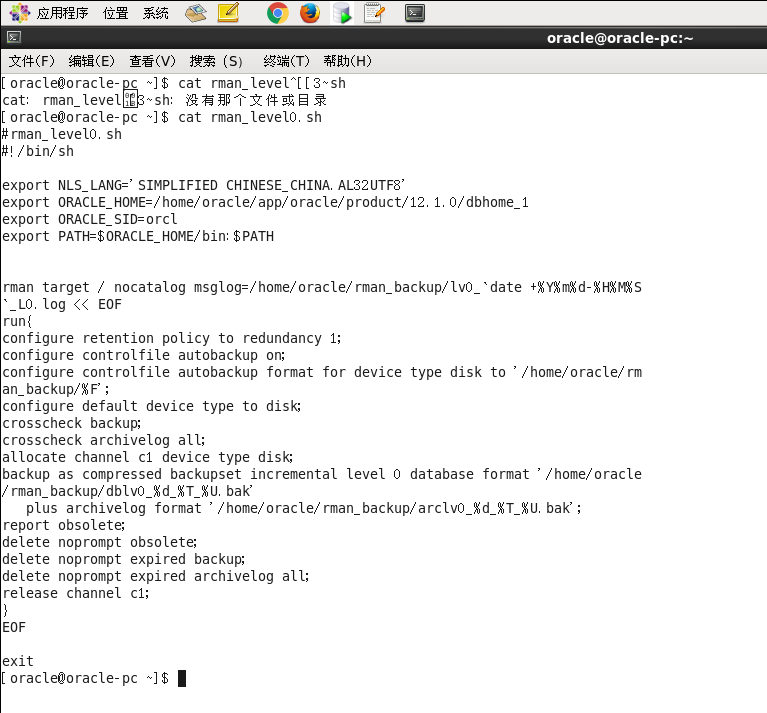
begin

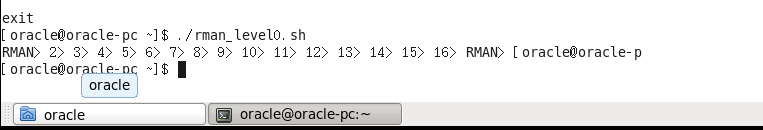
select seq\_admin.nextval

into :new.Adminid from dual;

end;

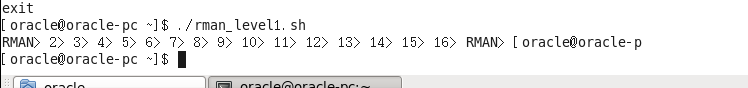
**5.开始全备份**：



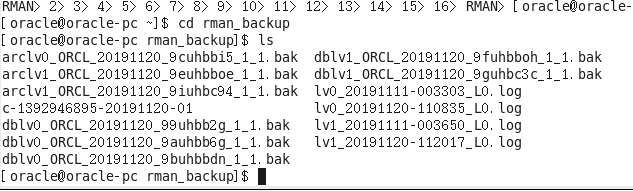


### 每天定时开始增量备份

![1@3%0@Y{%A8](CI[JN@6~@E](data:image/png;base64,)



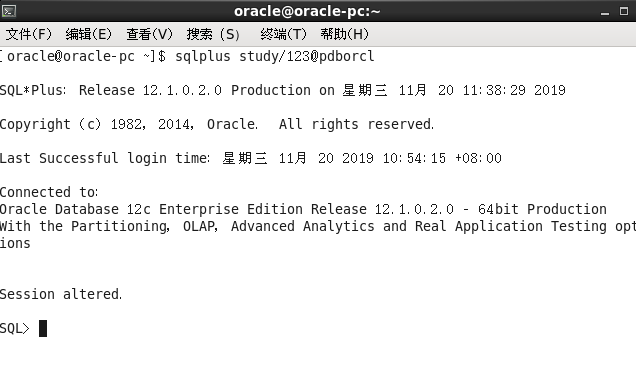
### 查看备份文件

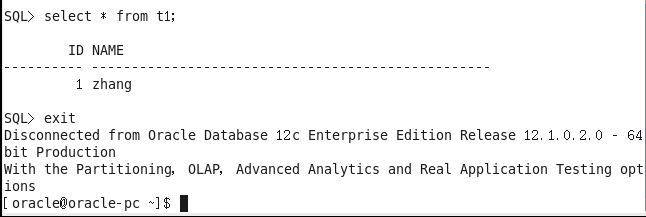


### 查看备份文件的内容

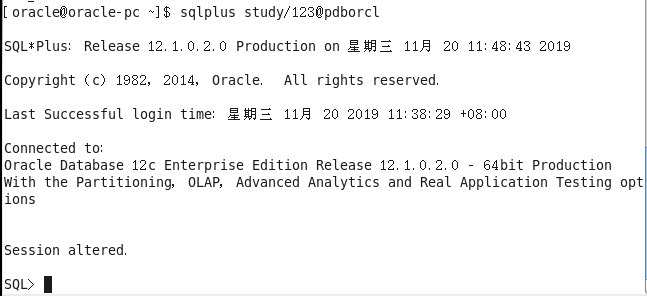


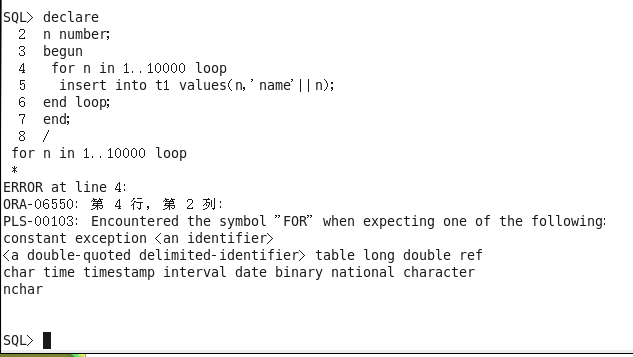
## 备份后修改数据



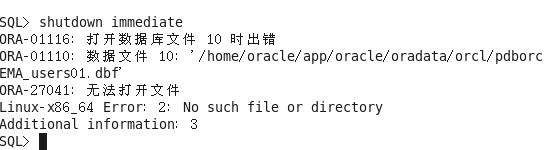


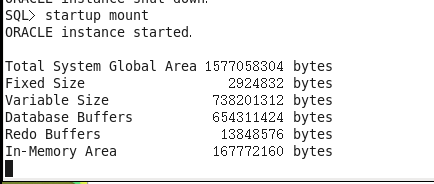
## 删除数据库文件，模拟数据库文件损坏



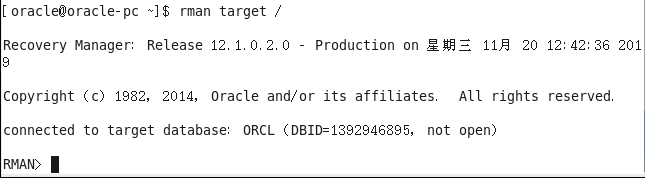


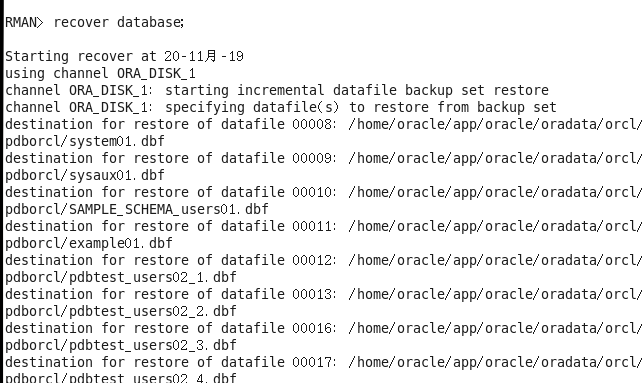
### 重启损坏的数据库到mount状态

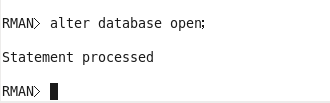




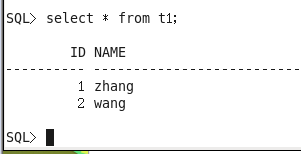
### 开始恢复数据库



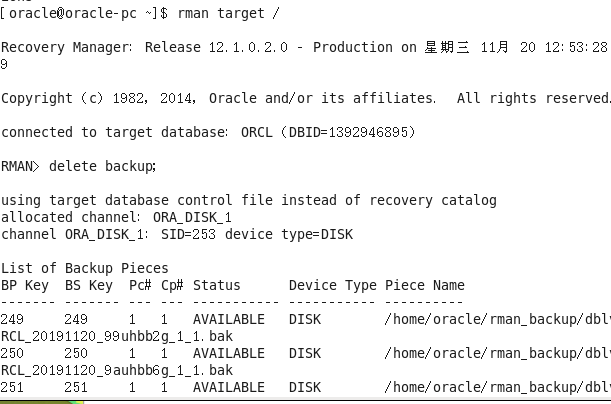


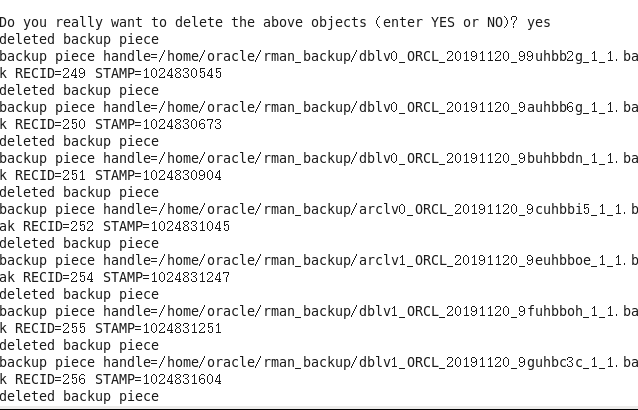


## 查询数据是否恢复



## 删除备份集





6.设计容灾方案

## 备库

mkdir -p /home/oracle/app/oracle/diag/orcl

mkdir -p /home/oracle/app/oracle/oradata/stdorcl/

mkdir -p /home/oracle/app/oracle/oradata/stdorcl/pdborcl

mkdir -p /home/oracle/arch

mkdir -p /home/oracle/rman

mkdir -p /home/oracle/app/oracle/oradata/stdorcl/pdbseed/

mkdir -p /home/oracle/app/oracle/oradata/stdorcl/pdb/

删除原有数据库:

$sqlplus / as sysdba

shutdown immediate;

startup mount exclusive restrict;

drop database;

启动到nomount

$sqlplus / as sysdba

startup nomount

## 主库:

$sqlplus / sysdba

select group#,thread#,members,status from v$log;

alter database add standby logfile group 5 '/home/oracle/app/oracle/oradata/orcl/stdredo1.log' size 50m;

alter database add standby logfile group 6 '/home/oracle/app/oracle/oradata/orcl/stdredo2.log' size 50m;

alter database add standby logfile group 7 '/home/oracle/app/oracle/oradata/orcl/stdredo3.log' size 50m;

alter database add standby logfile group 8 '/home/oracle/app/oracle/oradata/orcl/stdredo4.log' size 50m;

主库环境开启强制归档

ALTER DATABASE FORCE LOGGING;

编辑主库以及备库的/home/oracle/app/oracle/product/12.1.0/dbhome\_1/network/admin/tnsnames.ora文件

$gedit /home/oracle/app/oracle/product/12.1.0/dbhome\_1/network/admin/tnsnames.ora

ORCL =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.206.131)(PORT = 1521))

)

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = orcl)

)

)

stdorcl =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.206.132)(PORT = 1521))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SID = orcl)

)

)

在主库上生成备库的参数文件:

SQL>create pfile from spfile;

将主库复制到备库

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

## 备库

在备库上更改参数文件

$gedit /home/oracle/app/oracle/product/12.1.0/dbhome\_1/dbs/initorcl.ora

在备库增加静态监听

$gedit /home/oracle/app/oracle/product/12.1.0/dbhome\_1/network/admin/listener.ora

重新启动,备库开启实时应用模式:：

$sqlplus / as sysdba

shutdown immediate

startup

alter database recover managed standby database disconnect;

## 数据同步测试，主库+备库

在主库修改数据，在备库查询修改。 Select Name,Sequence#,applied,completion\_time From v$archived\_log Order By Sequence# Desc;

参考：https://blog.csdn.net/qq\_30105095/article/details/76849394