



综合

ORACLE 数据库入门



综合

一、Oracle 体系结构

(1)物理结构

datafiles

redo log files

control files

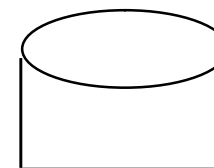
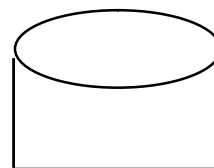
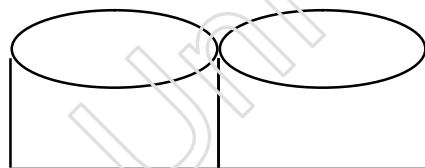
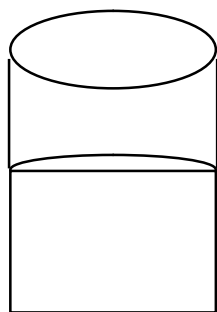
parameter file

数据文件

日志文件

控制文件

参数文件



data files

redo log files

control file

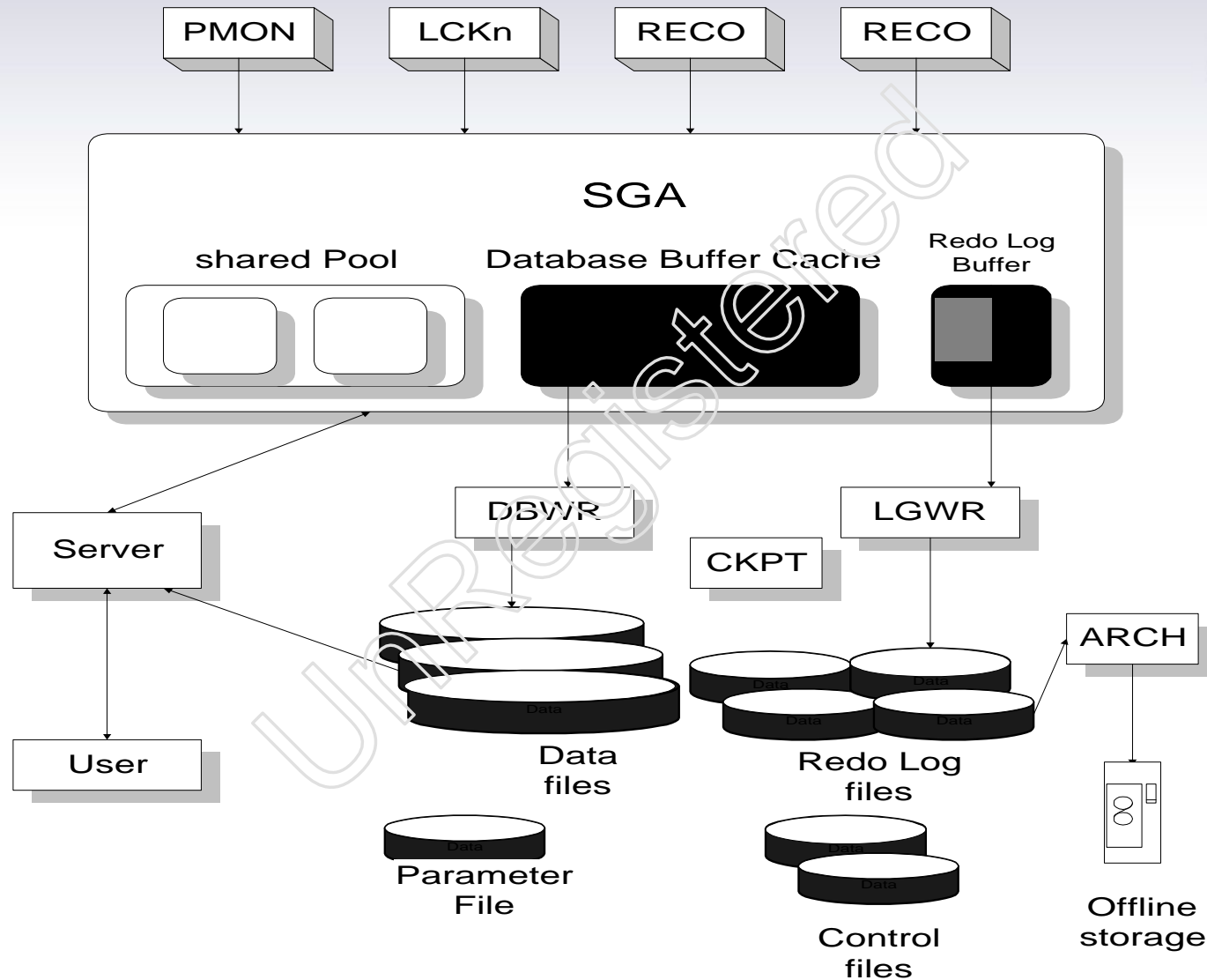
parameter files

*.dbf

*.log

*.ctl

Init.ora





综合

(2)内存结构(SGA)

占OS内存的60-70%，大小可由参数文件内参数计算

shared pool(共享池), database buffer cache(数据缓冲区),

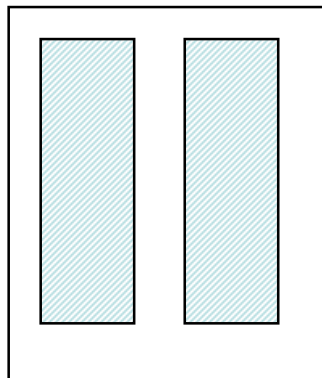
redo log buffer(重做日志缓冲区) (如以下图所示)

$$\text{SGA} = \text{share_pool_size} + \text{db_block_size} * \text{db_block_buffers} + \text{log_buffers}$$

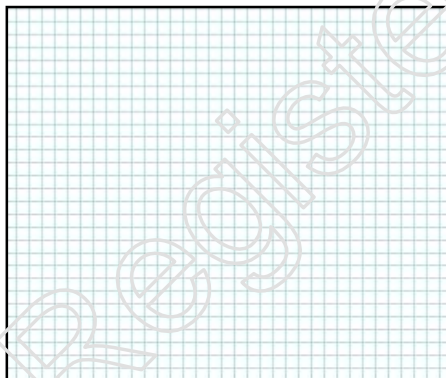


综合

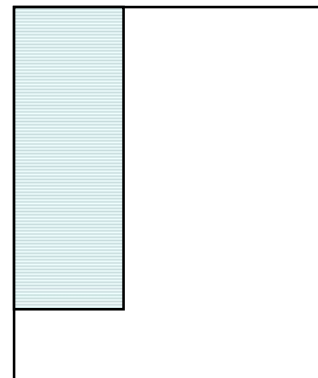
shared_pool



database_buffer_cache



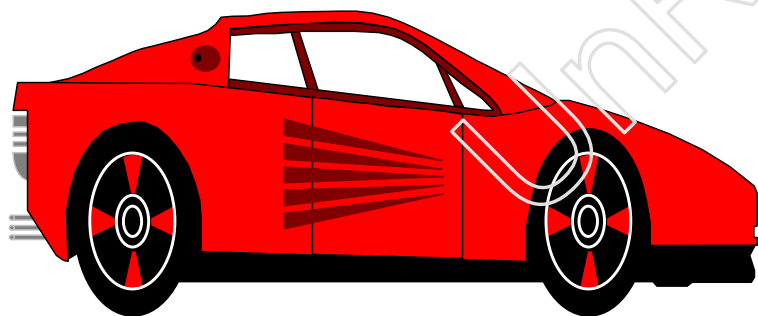
redo_log buffer



(3) instance=SGA+background process

实例 = 内存分配 + 一组后台进程

如果把Oracle比作一部汽车，instance相当于汽车的发动机一样，启动oracle前提应先启动instance.



SGA

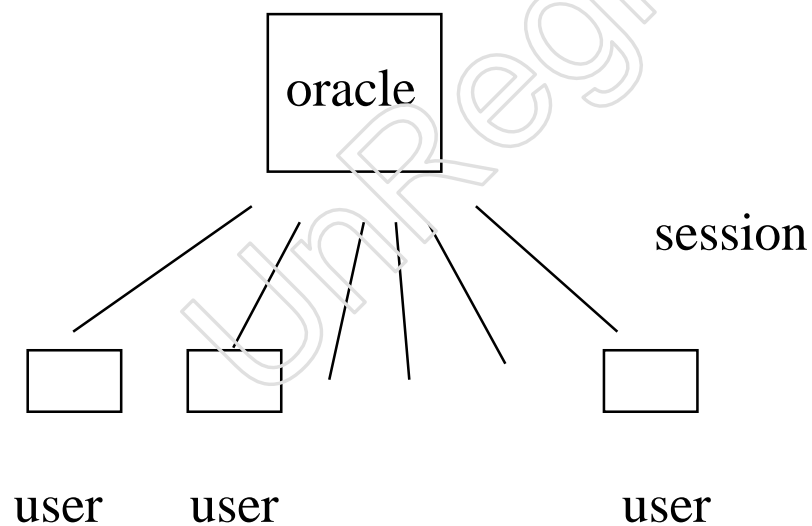
background
process



综合

(4)session(连接)

Oracle是多用户、多任务、可分布式管理的数据库，同时可有许 多个用户对数据库操作。





综合

(5)transaction(一组修改动作的集合) 交易事务

Eg:

1、

insert
delete
commit

DDL (数据定义语句)

例如: create,alter,drop,commit 等
每两个DDL语句间是一个transaction

2、

update

DML (数据控制语句)

rollback

例如: Insert,Delete,Update



综合

(6)后台进程 PMON, LCLN, RECO, SMON, DBWR, LGWR, CKPT, ARCH

PMON 做程序的清洁工作, 处理一些不正常退出的事件.

SMON 做系统的清洁工作, 执行系统出错后自动恢复工作.

LCKN Oracle系统表级或行级加锁的进程.

RECO 恢复进程.

DBWR 数据写进程

LGWR 日志文件写的进程

CKPT 检测点

ARCH 归档方式备份进程



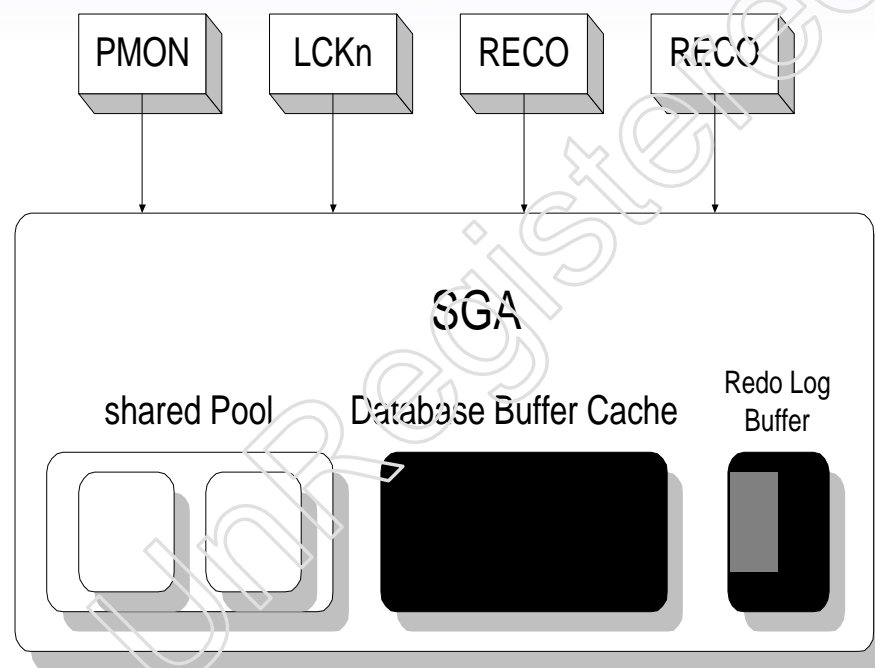
(7)分析一个SQL语句是怎样在Oracle内部工作的。

- A、用户发出SQL请求，打开游标；
- B、把SQL语句语法分析，执行计划，数据字典等信息存入内存中共享池内；
- C、从数据文件中把相关数据块读入数据缓冲区；
- D、做相应操作，若做修改，先加上行级锁，经确认后，把改过前后记录内容存入重做日志缓冲区内；
- E、返回结果给用户，关闭游标。

备注：SQL语句大小写敏感的，同样的一个语句，若大小写不同，oracle需分析执行两次，每句后必以“；”结束。



综合





综合

二、启动和关闭数据库。

(1)启动 %svrmgrl. 唤醒 SVRMGRL数据库管理

SVRMGRL > connect internal;以系统管理员身份登录。

SVRMGRL > startup

启动instance

连上数据库

打开数据库



综合

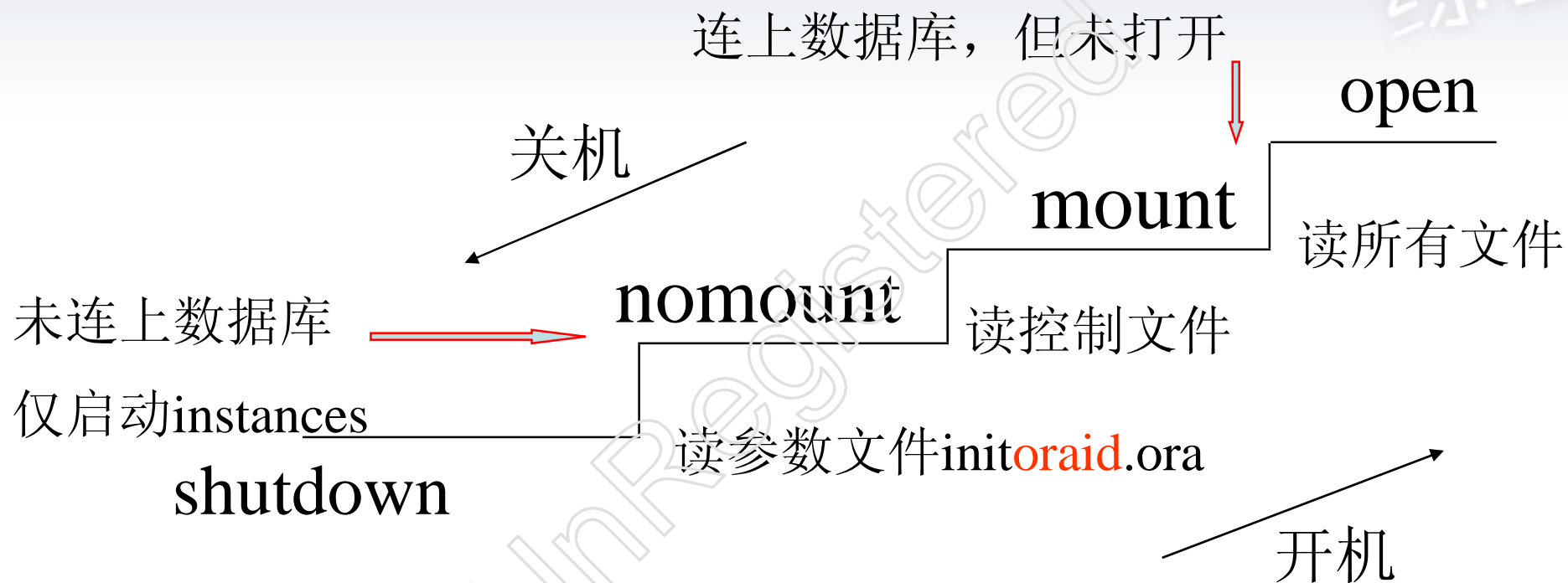
(2)关闭 (旧版Oracle用%sql>alter system set lmode=0 scope=spfile;)

%svrmgrl. (V7.3.2.0以上) 唤醒SVRMGRL状态

SVRMGRL>connect internal;

SVRMGRL>shutdown.

(3)启动和关闭Oracle数据库过程图。





综合

(4)如果不小心物理上删除了一Oracle的数据文件，比如说，某应用表空间所对应数据文件“adc.dbf?Oracle读控制文件时，和打开数据库时所面对的参数不一致，Oracle数据库将启动不了，解决这种问题的方法是把其对应的表空间先卸下，再删除，以保证控制文件描述和物理上存在文件一致。

```
%svrmgrl
```

```
SVRMGR>connect internal
```

```
SVRMGR>startup mount
```

```
SVRMGR>alter database datafile '/directory/abc.dbf'  
offline;
```

```
SVRMGR>alter database open;
```

```
SVRMGR>drop tablespace abc;
```



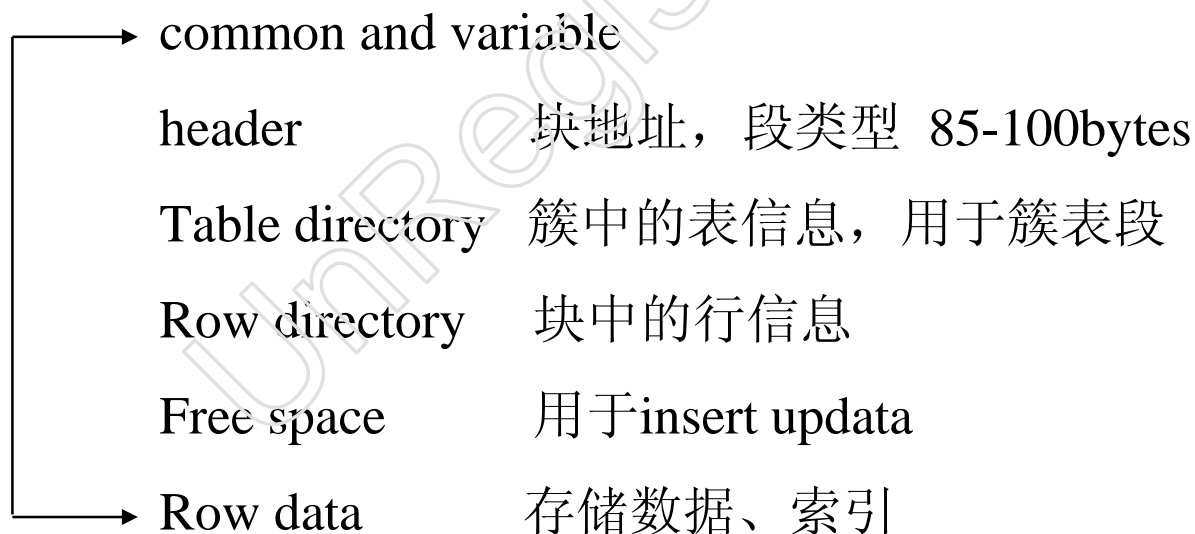

三、Oracle数据的存储结构

1、Oracle数据存储单位

- a、block 数据块： 2k 最小的I-O单位，伴随database产生而产生，不可变
- b、extent 一组连续的数据块： 是用户所能分配存储的最小单位
- c、segment 段： 有共同结构的一个或几个区域(extent)
- d、tablespace 表空间： 一组物理数据的逻辑组合，(象逻辑间数据仓库)
- e、file 文件： 属于某个表空间的物理文件
- f、database 数据库： 一组表空间所构成的逻辑的可共享的数据。

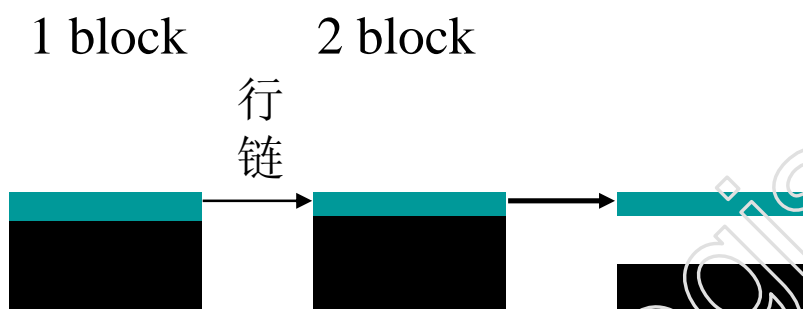
(系统管理员能在SQLDBA状态, 查看视图dba-extents,dba-segments, dba-tablespace,dba-data-files查看所有的extent,segment,tablespace和 datafile)

block
结构





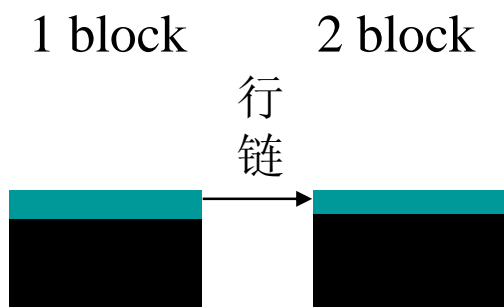
3、行链



当要存储的数据无法在一个数据块中存放时，需分配两个或多个数据块，标志这几个data block连接关系的存储信息



4、行迁移



当一个data block中的某部分数据经修改后增涨太快，无法继续放在本数据块中，从需把它搬迁到另一个data block，以优化存储结构，标志这个data block迁移的存储信息称为行迁移。



5、控制block空间使用的几个参数

- pctfree 20%左右 它们是互相消涨的
- pctused 40%左右
- Initrans 在单一块中最初活动的交易事务数
- Maxtrans 在单一块中最大交易事务数

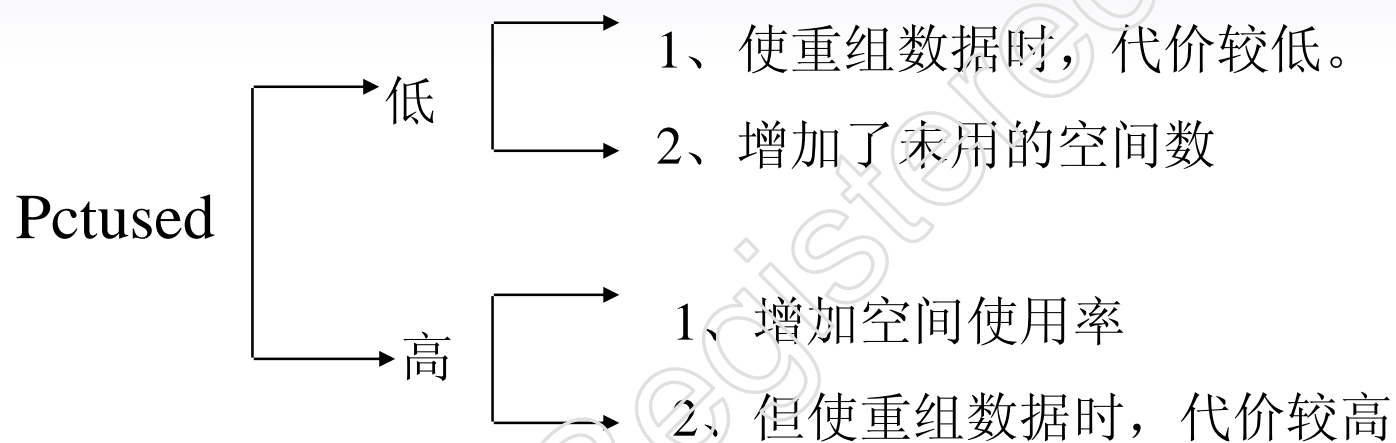


综合





综合





6、决定extent的参数

pctincrease

→ initial 最初分配的空间数(缺省为10k,5
个数据块)

next 下一步分配的空间数

maxextents 最大分配的extent数

minextents 最小分配的extnet数, 所有重
做日志回滚段的存储结构, 必
须成对地分配extent

pctincrease 增长率, 指数级增长,
$$\left(1 + \frac{\text{pctincrease}}{100}\right)^n$$

optimal 尽量设小, 或为0(缺省为空,
仅用于回滚段)

→ freelist



存储参数的设定规则:

- 1、在对象级的存储参数设置值覆盖表空间级的设置
- 2、未在对象级设置存储参数，由表空间级数设置决定
- 3、未在表空间级设置存储参数，由Oracle数据库级参数设置决定
- 4、若存储参数改变后，新的选项只针对未分配的extents有效。



7、segment 段的分类

data 数据段 存储对象 object(table,view,indexsequence...)
index 索引段
temporary 临时段 用做(join,group by,order by sorting操作)
rollback 回滚段 用于记录修改前后信息, minextent为2,
increace为0
bootstrap 启动段 存储数据字典系统信息
不能读写, 放在系统表空间内, 约占40
几个block



综合

data segment中可能有十种不同类型的存储对象

- (1)table (2)view 实为一个select语句
- (3)index(一个表不多于3个) (4)cluster (簇) (5)sequence 序列
- (6)synonme 用于定义某个 远程数据库同义词，实现分式数据库管理透明
- (7)snapsot 快照 (8)stored procedure(function)
- (9)package 程序包
- (10)db trigger 数据触发器，处理update,delete,insert中可能出现的问题
见 sun屏幕，创建.....
SQL语句以 捰糅结束
缺省：10分钟无SQL请求，做rollback
缺省：连接在线connect time (1个小时)



8、table存储，按行格式

1	2	3	4	5	6
---	---	---	---	---	---	--------

- 1、rowoverhead 行头
- 2、row piece 列数
- 3、cluster key (不一定有)
- 4、rowid 行的唯一标记(有行链时，记录其它block的rowid)
- 5、length
- 6、value



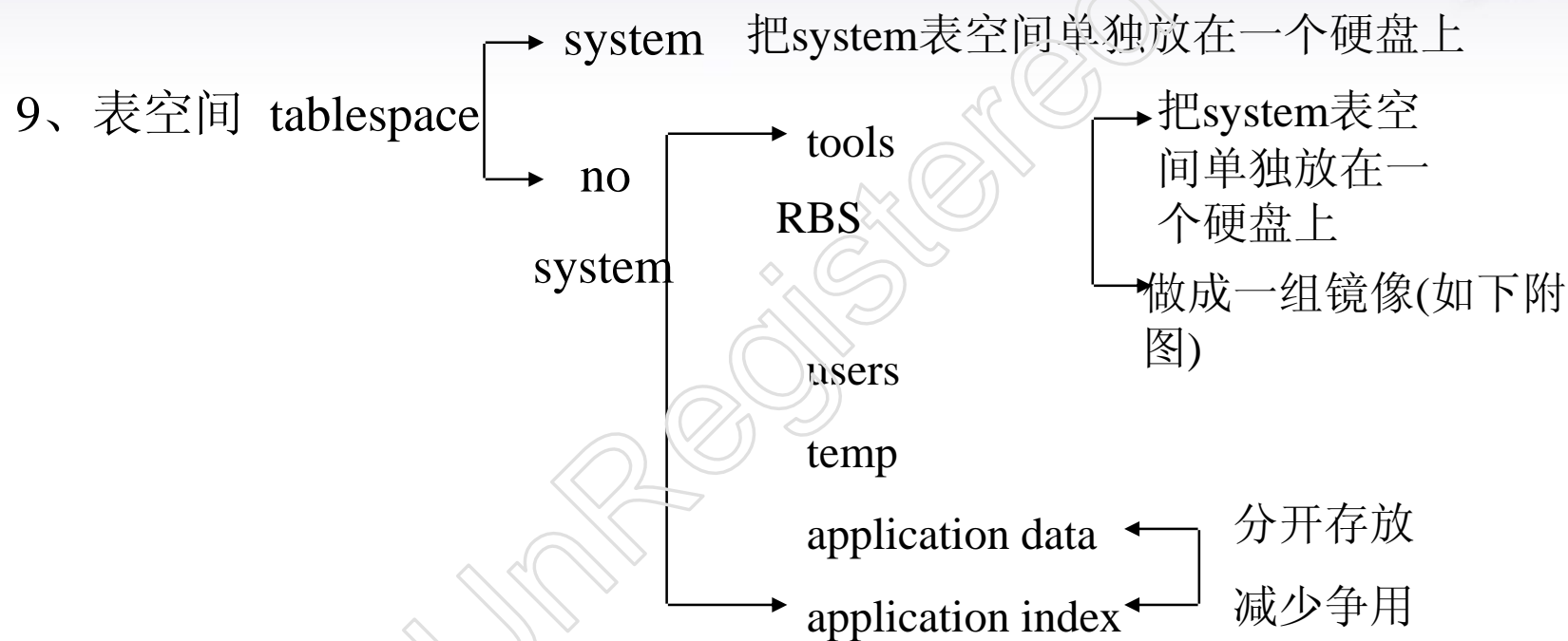
综合

当字段值为null时，length为0，无6字value部分，
所以某字段常为空，应放table最后；尽量不设为
null的字段

rowid: xxxxxxxx . x x x x . xxxx

 ↓ ↓ ↓

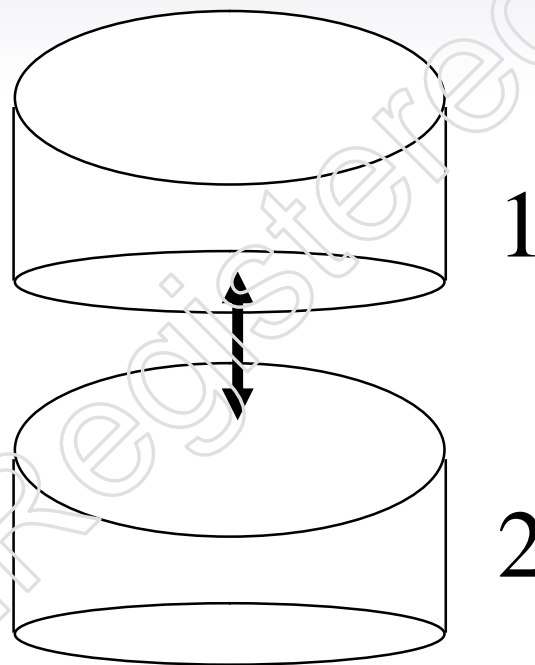
 块物理地址 该row在块中的行号 块所属的文件号





附图:

(rollback)





四、备份和恢复

1、备份分类

(1)逻辑备份 用Export/Import 实用工具

(2)物理备份 操作系统下的备份

- 归档方式 不关闭database状态
- 不归档方式 关闭database后，把物理文件进行备份
(cp tar)

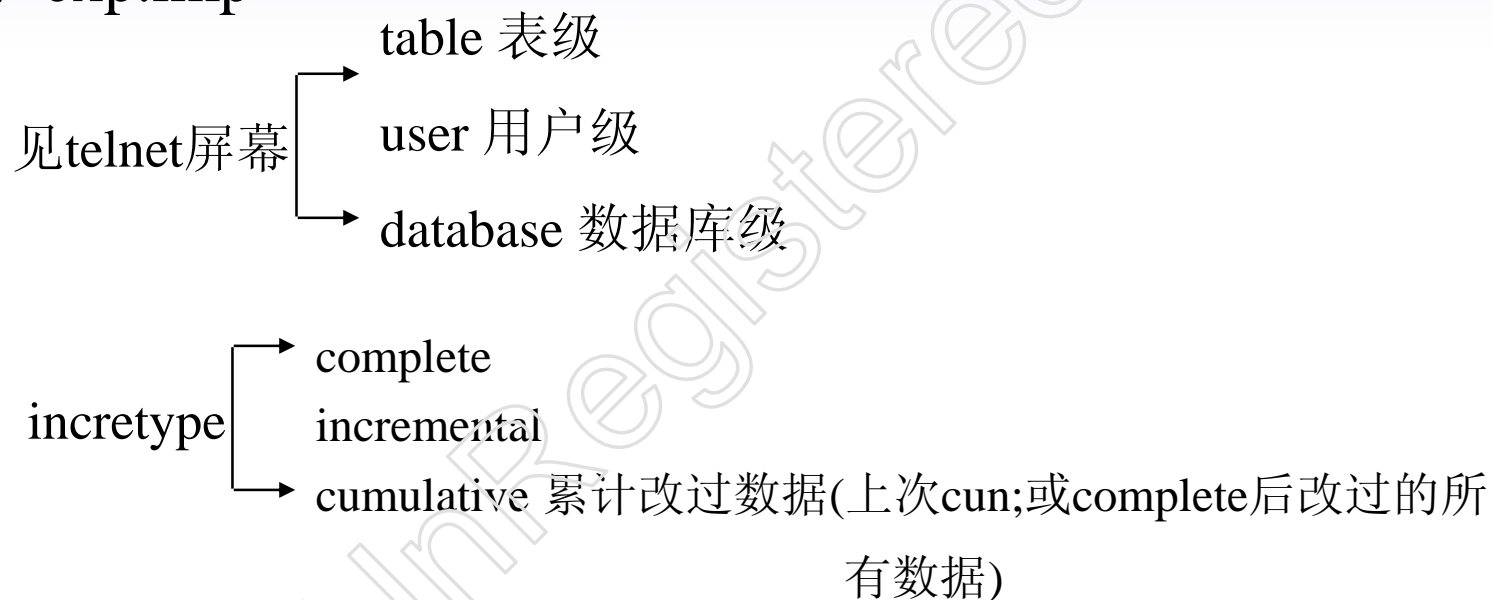


2、SVRMGR>archire log list 看当前数据库归档方式



综合

3、exp.imp





imp步骤:

- (1)create table
- (2)insert data
- (3)create index
- (4)create triggers,constraints

综合



综合

4、备份和恢复策略

逻辑备份仅能恢复到上次的备份点

物理备份中的不归档备份可恢复到上次备份点

物理备份中的归档备份恢复到failure的前一刻

见P6-98逻辑备份日刻表



恢复: (1)imp system/manager file=incr19.dmp
inctype=system 恢复最后状态信息

(2)imp system/manager file=x1.dmp incype=restore

(3)imp system/manager file=c8.dmp incype=restore

(4)imp system/manager file=c15.dmp incype=restore

(5)imp system/manager file=c16.dmp incype=restore

(6)imp system/manager file=c17.dmp incype=restore

(7)imp system/manager file=c18.dmp incype=restore

(8)imp system/manager file=c19.dmp incype=restore



综合

五、用户管理

1、创建、改变、删除和监控oracle的新用户



综合

(1) 赋予合法的用户或和密码

SVRMGL>CREATE USER username IDENTIFIED BY userpassword

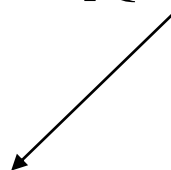
[DEFAULT TABLESPACE tablespace1]

[TEMPORARY TABLESPACE tablespace2]

[QUOTA n K ON tablespace1];

M

UNLIMITED



空间限额



综合

(2)授于用户连接Oracle数据库的权限

```
SVRMGL>grant  connect to  username [with grant option];  
                privilege..... rolename
```



综合

常用的几个角色role (权限的集合)

- connect (8) 连上Oracle,做最基本操作
- resource(5) 具有程序开发最基本的权限
- dba (77)数据库管理员所有权限
- exp-full-database 可把数据库整个备份输出的权限
- imp-full-database 可把数据库整个备份恢复输入的权限

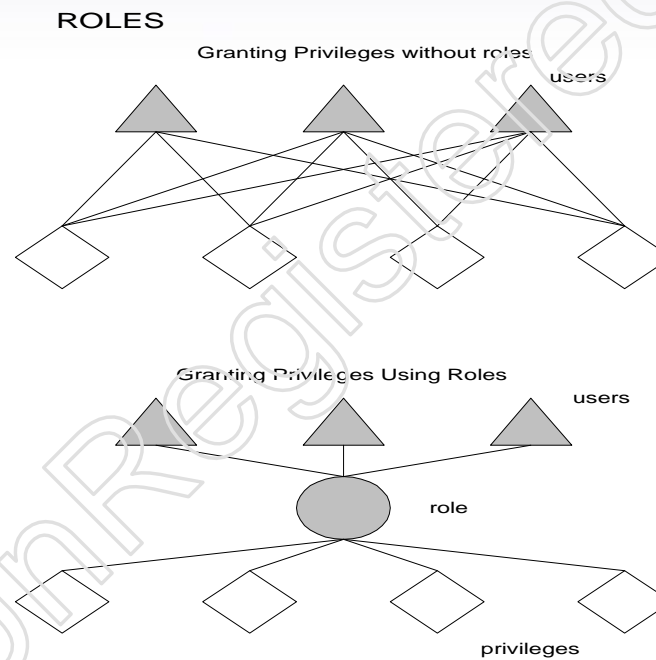


综合

- (3)alter user;
- (4)drop user [cascade];
- (5)revoke role from user;



2、权限的管理





权限的分类

- system privilege 针对整个系统操作的权限
- object privilege 针对整个具体object操作的权限

可查看

- dba-sys-privs,
- dba-fab-privs;视图
- dba-col-privs;



综合

六、SQL LOADER

见屏幕 /oracle/home/rdbms/demo/ulcase*.ctl

usend=scott/tiger