补充内容：

oracle 10g 卸载篇

oracle 10g 基础篇

1.创建表

方式一：

create table dept

(

deptno number(2) not null,

dname varchar2(13),

constraint pk\_dept primary key(deptno)

using index tablespace rensh\_app

storage-----------------------------指定块的存储参数

(

initial 64k ----------------------起始扩展区大小

next 5k---------------------------下一个扩展区大小

minextends 1----------------------可为段分配总的最小区的个数

maxextends 21474836645------------可为段分配总的最大区的个数

pctextends 1

freelists 1 freelistgroups 1----用于手工段空间管理MSSM,多重的freelists,freelistgroups可提高并发能力

)

pctfree 10-------------------------用于自动段空间管理ASSM,参数有pctfree、buffer\_pool、initrans、maxtrans

initrans 2-------------------------进行更新操作的最小和最大事务数

maxtrans 255-----------------------oracle 10g 已失效

)

tablespace

rensh\_app

pctfree 10

initrans 1

maxtrans 255

pctused 40---------------------------用于手工段空间管理MSSM,多重的freelists,freelistgroups可提高并发能力

storage

(

initial 64k

next 0k

minextends 1

maxextends 21474836645

pctincrease 0---------------------指定扩展区递增的百分比，新的next=(1+pctincrease/100)\*old\_next 如果为0，自由区不会自动合并，会产生碎片

freelists 1 freelistgroups 1

)

logging;----------------------------重做日志

方式二：

create table emp\_infor as select \* from emp where sal>2000;

2.修改表

修改表的存储参数 alter table emp storage(next 100k);

使表上约束生效 alter table emp\_infor enable constraint fk\_emp\_deptno;

使表上约束失效 alter table emp\_infor diable constraint fk\_emp\_deptno;

给表添加一个校验约束 alter table emp add constraint ck\_sal check(sal>1100);

表更名 alter table emp\_infor rename to emp\_if;

增加一个主键 alter table emp\_infor add constraint pk\_emp\_id primary key(empid) using index tablespace pk\_index storage(initial 100k next 100k);

增加列 alter table emp add email\_address varchar2(20);

修改列名 alter table emp rename column sal to salary;

修改列属性 alter table emp modify email varchar2(16);

删除列 alter table emp\_infor drop column email\_address;

删除表 drop table employee;

删除表及约束 drop table emp cascade constraints;

使用select在默认表空间上创建一个空表 create table student\_infor tablespace mystudent as select \* from student where 1=2;

使用select创建带有数据及全部定义的表 create table student\_infor tablespace mystudent as select \* from student;

并行参数parallel指定一个命令可并行执行的数量

create table tele\_code

(

tele\_code varchar2(5),

owner varchar2(2),

city varchar2(40),

parallel (degree 9)

) tablespace myteleapp;

为表添加注释 comment on table tele\_code is 'dalian city telecomservice';

为列添加注释 comment on column emp.sal is 'the data is employee salary';

查看表的注释 select table\_name,comments from user\_tab\_comments;

查看列的注释 select table\_name,column\_name,comments from user\_col\_comments;

将列设为不可用 alter table employee set unused column job;

删除表中不可用的列 alter table test drop unused columns;

删除大表中的列 alter table test dorp column fourth\_col checkpoint 1000;

将表移动到不同表空间 alter table emp move tablespace users;

将含有lob段的表移动到不同表空间

alter table empinor move

tablespace human\_tbs

lob (emp\_image) store

as lob\_emp\_image (tablespace enterprise\_tbs);

分配未使用空间 alter table employee deallocate unused;

更改表的logging为nologging alter table employee nologging;

3.表的类型

常规表：用create table 语句创建的表就是堆表

全局临时表：总是在用户的临时表空间中创建，包含特定的会话数或者特定的事务数据

创建全局临时表

create global temporary table session\_temp\_table

(

student number primary key,

stdname varchar2(30),

deptno number

)

on commit prserve rows;---会话

delete ---事务

索引组织表：以B树结构存储行的主键和数据的一种存储方式

使用情况

a)当表中若干个列组成主关键字且在一行中占教大部分时

b)开发大型数据库或联机事务处理

c)开发时间系列的应用

d)存储嵌套表

create table emp\_work

外部表

分区表

簇表

嵌套表

对象表

xml表

oracle 10g 安全篇

1.限定IP访问数据库

在<oracl\_home>\network\admin\sqlnet.ora中加入以下内容

tcp.validnode\_checking=yes;//是否开启

tcp.invited\_nodes=(gps.wzs.zj,192.168.0.1);//允许访问的IP 可以是IP或域名，多个以逗号隔开

tcp.excluded\_nodes=(hl.wzs.zj,127.0.0.1);//不允许访问的IP

oracle 10g 提高篇

1.多表合并

背景：将两个或两个以上表的数据合并到一个表中，如果表中已经存在，则更新；如果不存在，则插入。

merge into target\_table [t\_alias]

using org\_table [t\_alias] on (condition)

when matched then <merge\_update\_clause>

when not matched then <merge\_insert\_clause>;

update或insert子句是可选的；

可以进行有条件的更新操作；

可使用where子句完成有条件的插入操作；

支持在on条件上的常量过滤谓词，将所有行插入到目标表中而无需将两个表进行连接；

在完成更新期间，可选的delete子句可删除行

例子1：

bonuses 奖金表

employees 职员表

其中，bonuses已经含有销售人员数据

人力资源部决定为所有雇员颁发奖金，非销售人员的奖金是其薪水的1%；销售人员的奖金是在其原来奖金基础上加上薪水的1%

merge into bonuses d

using (select employee\_id,salary,department\_id from employees) s

on (d.employee\_id=s.employee\_id)

when matched then

update set d.bonus=d.bonsons\*0.01

when matched then

insert (d.employee\_id,d.bonus) values(s.employee\_id,s.salary\*0.1)

例子2:

products newproducts(product\_id,product\_name,category)

省略update或insert

merge into products p

using newproducts np on(p.product\_id=np.product\_id)

when matched then

update set p.product\_name=np.product\_name,p.category=np.category

merge into products p

using newproducts np on(p.product\_id=np.product\_id)

when not matched then

insert (p.product\_id,p.product\_name,p.category) values (np.product\_id,np.product\_name,np.category)

有条件的更新或插入

merge into products p

using newproducts np on(p.product\_id=np.product\_id)

where matched then

update set p.product\_name=np.product\_name,p.category=np.category where p.category='DVD'

where not matched then

insert (p.product\_id,p.product\_name,p.category) values (np.product\_id,np.product\_name,np.category) where np!='BOOKS'

无条件插入

merge into products p

using newproducts np on (1=0)

where not matched then

insert (p.product\_id,p.product\_name,p.category) values (np.product\_id,np.product\_name,np.category) where np!='BOOKS'

新的delete子句

merge into products p

using newproducts np on(p.product\_id=np.product\_id)

where matched then

update set p.product\_name=np.product\_name,p.category=np.category

delete where p.category='DVD'

where not matched then

insert (p.product\_id,p.product\_name,p.category) values (np.product\_id,np.product\_name,np.category)

2.单表分解

需要将一个表中的数据按照条件分别一次性地插入到多个表中

insert all|first

[when <condition-1> then]

insert <table1>[(col1,col2,……)]

[values(val1,val2,……)]

[when <condition-2> then]

insert <table2>[(col1,col2,……)]

[values(val1,val2,……)]

……

……

[else

into<table-n>[(col1,col2,……)]

[values(val1,val2,……)]

]

subquery-definition;

例子：针对销售总量来区分不同额度的订单，需要建立4个不同的表，并将不同数量级别的订单数据载入到这4个表中

small\_orders、medium\_orders、large\_orders、special\_orders(order\_id,customer\_id,order\_total,sales\_rep\_id)

insert all

when ott1<10000 then

insert small\_orders values(oid,ott1,sid,cid)

when ott1>10000 and ott1<20000 then

insert medium\_orders values(oid,ott1,sid,cid)

when ott1>20000 then

insert large\_orders values(oid,ott1,sid,cid)

when ott1>29000 and ott1<20000 then

insert special\_orders values(oid,ott1,sid,cid)

select o.order\_id oid,customer\_id cid,o.order\_total ott1,o.sales\_rep\_id sid,c.credit\_limit cl,c.cust\_email cem

from orders o,customers c

where o.customer\_id=c.customer\_id

3.分区表

分区方法:

范围分区(Range Partitioning)

散列分区(Hash Partitioning)

列表分区(List Partitioning)

复合范围-散列分区(Composite Range-Hash Partitioning)

复合范围-列表分区(Composite Range-List Partitioning)

范围分区

create table sales(invoice\_no number,

sale\_year int not null,

sale\_month int not null,

远程备份数据库

exp lctest/lc1212@/220.189.246.90:1521/orcl file=e:/lc.dmp

查看建表语句

select dbms\_metadata.get\_ddl('TABLE','PHONE\_BILL\_RECORD') from dual;

结果显示不全

set long 100000

set pages 0

select \* from dba\_tab\_partitions;

select \* from dba\_tab\_subpartitions;

oracle 字符集

select userenv('language') from dual;

函数

查看函数源码

select \* from user\_source t where t.name=upper('first\_digit');

调用java 类

1.用户必须拥有javauserpriv权限

2.创建java类含有静态方法

public class F2C{

public static double f2c(double fc){

return (5.0/9.0)\*(fc-32.0);

}

}

3.上传该编译过的java类到用户下

loadjava -user test/test F2C.java

4.创建对应的函数

create or replace function f2cjava(fc number)

return number as language java name 'F2C.show(double) return double';

也可以

create or replace and compile java source named f2c as

public class F2C{

public static double f2c(double fc){

return (5.0/9.0)\*(fc-32.0);

}

}

删除

drop java source f2c

包

查看包的基本信息

select \* from user\_arguments t where t.package\_name='P\_OVERLOAD';

查看包源码

select \* from user\_source t where t.name='P\_OVERLOAD';

表空间

dba\_data\_files,dba\_free\_space

查看所有表空间

select \* from dba\_data\_files;

表空间使用率

select total.tablespace\_name,round(total.MB,2) as Total\_MB,round(total.MB-free.MB, 2) as Used\_MB,round((1-free.MB/total.MB)\*100, 2) as Used\_Pct

from

(select tablespace\_name, sum(bytes)/1024/1024 as MB from dba\_free\_space group by tablespace\_name) free,

(select tablespace\_name, sum(bytes)/1024/1024 as MB from dba\_data\_files group by tablespace\_name) total

where free.tablespace\_name=total.tablespace\_name

休眠

a.授权

grant execute on dbms\_lock to user;

b.使用

dbms\_lock.sleep(5); 秒

快照(用于数据同步)

tbl\_user tbl\_user\_temp

将tbl\_user的数据同步到tbl\_user\_temp中

--日志

create snapshot log on tbl\_user;

--快照

create snapshot sn\_tbl\_user as select \* from tbl\_user;

--可能没有权限，授权

grant create snapshot to test;

--同步频率

alter snapshot sn\_tbl\_user refresh fast start with sysdate+30/(24\*60\*60) next sysdate+30/(24\*60\*60) with primary key;

--删除日志

drop snapshot log on tbl\_user;

--删除快照（物化试图）

drop materialized view sn\_tbl\_user;

--查看上次同步时间

select \* from all\_snapshot\_refresh\_times;

--查看下次同步时间

select \* from user\_jobs;

触发器

create or replace trigger TRI\_SN\_TBL\_USER\_SYNC

after insert on sn\_tbl\_user for each row

begin

if inserting then

insert into tbl\_user\_temp(id,name) values(:new.id,:new.name);

end if;

end;

oracle内置函数

对数值操作

abs(n):绝对值

select abs('-3') from dual;-----结果3

ceil(n):返回>=n的最小的整数

select ceil(3.1) from dual;-----结果4

floor(n):返回<=n的最大的整数

select floor(3.1) from dual;-----结果3

mod(m,n):求余，若n=0,返回m

select mod(5,2) from dual;-----结果1

power(m,n):求m的n次方的值

select power(2,3) from dual;-----结果8

round(m,n):四舍五入，n小数点后保留几位，默认为0

select round(3.25) from dual;-----结果3

select round(3.25,1) from dual;-----结果3.3

sqrt(n):平方根

select sqrt(25) from dual;-----结果5

sign(x):x<0,返回-1；x=0,返回0；x>0,返回1

select sign(-100) from dual;-----结果-1

select sign(0) from dual;-----结果0

select sign(100) from dual;-----结果1

对字符操作

ascii(char):返回ascii码表字符对应的数值十进制表示

select ascii('A') from dual;-----结果65

initcap(char):把每个字符串的第一个字符换成大写

select initcap('mr.dog') from dual;-----返回Mr.Dog

select initcap('mr dog') from dual;-----返回Mr Dog

lower(char):大写换成小写

select lower('Mr Dog') from dual;-----返回mr dog

upper(char):小写换成大写

select upper('Mr Dog') from dual;-----返回MR DOG

replace(char,str1,str2):字符串中所有str1换成str2

select replace('scotts','s','boy') from dual;-----返回boycottboy

substr(char,m,n):取出从m字符开始的n个字符的子串,默认n为取到末尾

select substr('world',2) from dual;-----返回orld

select substr('world',2,2) from dual;-----返回or

length(char):长度

select length('world') from dual;-----返回5

||:并置运算符

select 'wor'||'ld' from dual;-----返回world

concat(str1,str2):连接字符串

select concat('wo','rld') from dual;-----返回world

lpad(char,n,char2):用char2填充char的左边，让char的长度等于n,如果char的长度大于n,则去char的前n位

select lpad('world',10,'\*') from dual;-----返回\*\*\*\*\*world

select lpad('world',2,'\*') from dual;-----返回wo

ltrim(string,trim\_set):删除左边的字符，trim\_set要去除的字符集合

select ltrim(' world') from dual;-----返回world

select ltrim('world','w') from dual;-----返回orld

rpad(char,n,char2):用char2填充char的右边，让char的长度等于n,如果char的长度大于n,则去char的前n位

select rpad('world',10,'\*') from dual;-----返回world\*\*\*\*\*

select rpad('world',2,'\*') from dual;-----返回wo

rtrim(string,trim\_set):删除右边的字符，trim\_set要去除的字符集合

select rtrim('world ') from dual;-----返回world

select rtrim('world','d') from dual;-----返回worl

instr(string,set,start,n):字符串string中从start位置开始第n次出现set的位置

select instr('aptech is aptech or not aptech','ap',1,3) from dual;-----返回25

translate(str,str1,st2):str字符串中用str2相应位置的字符取代str1字符

select translate('abca','ab','BA') from dual;-----结果BAcB

对日期操作

sysdate:返回当前日期和时间

select sysdate from dual;

last\_day(date):返回本月的最后一天

select last\_day(to\_date('2012-02-01','yyyy-mm-dd')) from dual;-----返回2012/2/29

add\_months(date,n):date日期后推迟n个月

select add\_months(to\_date('2012-02-01','yyyy-mm-dd'),8) from dual;-----返回2012/10/1

decode(value,if1,then1,if2,then2,...,else)：当value的值为if1时，返回then1;当value的值为if2时，返回then2;否则返回else

wm\_concat(colname):列转行

sqlplus中执行sql文件

@+sql文件路径

如

@D:\Soft\oracle\product\10.2.0\db\_1\sqlplus\admin\plustrce.sql

explain

查看执行计划有3种

1.

set auto trace on;

报错

执行ORACLE\_HOME\sqlplus\admin\plustrce.sql,创建plustrce角色

2.

explain plan select \* from tbl\_user;

select \* from table(dbms\_xplan.display) 或 select plan\_table\_output from table(dbms\_xplan.display('plan\_table'))