# Oracle队列实现

-- 核心技术点：for update

## 创建测试表

create table t

( id number primary key,

processed\_flag varchar2(1),

payload varchar2(20)

);

## 创建函数索引

create index

t\_idx on

t( decode( processed\_flag, 'N', 'N' ) );

插入几条测试数据

insert into t

select r,

case when mod(r,2) = 0 then 'N' else 'Y' end,

'payload ' || r

from (select level r

from dual

connect by level <= 5)

/

## 方式一，通过函数返回未锁定行

### 创建队列获取一行数据的函数

支持Oracle8.0及以后的版本

create or replace

function get\_first\_unlocked\_row

return t%rowtype

as

resource\_busy exception;

pragma exception\_init( resource\_busy, -54 );

l\_rec t%rowtype;

begin

for x in ( select rowid rid

from t

where decode(processed\_flag,'N','N') = 'N')

loop

begin

select \* into l\_rec

from t

where rowid = x.rid and processed\_flag='N'

for update nowait;

return l\_rec;

exception

when resource\_busy then null;

when no\_data\_found then null;

end;

end loop;

return null;

end;

/

### 获取未加锁的第一行数据

declare

l\_rec t%rowtype;

begin

l\_rec := get\_first\_unlocked\_row;

dbms\_output.put\_line( 'I got row ' || l\_rec.id || ', ' || l\_rec.payload );

end;

/

eoda/muphy> I got row 2, payload 2

### 获取未加锁的第二行数据

declare

pragma autonomous\_transaction;

l\_rec t%rowtype;

begin

l\_rec := get\_first\_unlocked\_row;

dbms\_output.put\_line( 'I got row ' || l\_rec.id || ', ' || l\_rec.payload );

commit;

end;

/

eoda/muphy> I got row 4, payload 4

## 方式二，直接通过skip locked实现

### 获取未加锁的第一行数据

declare

l\_rec t%rowtype;

cursor c

is

select \*

from t

where decode(processed\_flag,'N','N') = 'N'

FOR UPDATE

SKIP LOCKED;

begin

open c;

fetch c into l\_rec;

if ( c%found )

then

dbms\_output.put\_line( 'I got row ' || l\_rec.id || ', ' || l\_rec.payload );

end if;

close c;

end;

/

eoda/muphy> I got row 2, payload 2

### 获取未加锁的第二行数据

declare

pragma autonomous\_transaction;

l\_rec t%rowtype;

cursor c

is

select \*

from t

where decode(processed\_flag,'N','N') = 'N'

FOR UPDATE

SKIP LOCKED;

begin

open c;

fetch c into l\_rec;

if ( c%found )

then

dbms\_output.put\_line( 'I got row ' || l\_rec.id || ', ' || l\_rec.payload );

end if;

close c;

commit;

end;

/

eoda/muphy> I got row 4, payload 4

--参考自Oracle编程艺术 深入理解数据库体系结构第三版