

17

Memonitor dan Menangani Lock Conflict

By: Ahmad Syauqi Ahsan

ORACLE®

TUJUAN

Setelah menyelesaikan bab ini, anda seharusnya dapat melakukan hal-hal berikut:

- Mendeteksi dan Mengatasi masalah Lock
- Mengatur deadlock

LOCKS

- Mencegah multiple session dari proses mengubah data yang sama pada waktu yang sama
- Secara otomatis mendapatkan level serendah mungkin untuk statement yang diberikan

Transaction 1



Transaction 2



```
SQL> UPDATE hr.employees  
2 SET salary=salary+100  
3 WHERE employee_id=100;
```

```
SQL> UPDATE hr.employees  
2 SET salary=salary*1.1  
3 WHERE employee_id=100;
```

Mekanisme Locking

- **Level tertinggi dari concurrency data**
 - Baris level penguncian untuk untuk inserts, update, delete
 - Tidak ada lock yang dibutuhkan untuk antrian
- **Manajemen otomatisasi antrian**
- **Penguncian berlangsung sampai transaksi berakhir (dengan operasi commit atau rollback)**

Transaction 1



Transaction 2



```
SQL> UPDATE hr.employees  
2 SET salary=salary+100  
3 WHERE employee_id=100;
```

```
SQL> UPDATE hr.employees  
2 SET salary=salary*1.1  
3 WHERE employee_id=101;
```

Data Concurrency

Time: 09:00:00	Transaction 1	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=100;
	Transaction 2	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=101;
	Transaction 3	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=102;

	Transaction x	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=xxx;

DML Lock

Transaction 1

```
SQL> UPDATE employees  
 2 SET salary=salary*1.1  
 3 WHERE employee_id= 107;  
1 row updated.
```

Transaction 2

```
SQL> UPDATE employees  
 2 SET salary=salary*1.1  
 3 WHERE employee_id= 106;  
1 row updated.
```

Setiap transaksi DML harus mendapatkan 2 lock :

- row-exclusive Lock untuk baris yang sedang diupdate
- Shared table-level Lock untuk tabel yang berisi banyak baris


Mekanisme Antrian

Sebuah mekanisme antrian menjaga jalur dari:

- **Session yang menunggu untuk dikunci**
- **Permintaan mode kunci**
- **Perintah dimana session meminta penguncian**

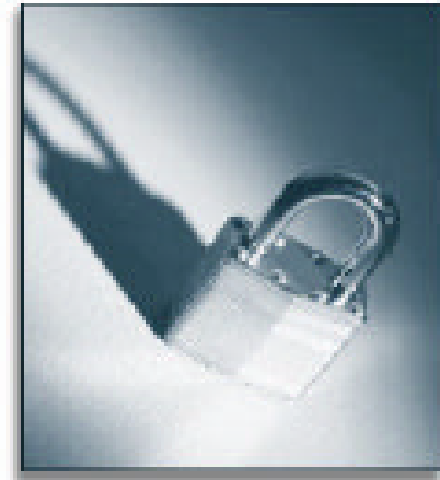


Lock Conflicts

Transaction 1	Time	Transaction 2
UPDATE hr.employees SET salary=salary+100 WHERE employee_id=100; 1 row updated.	9:00:00	UPDATE hr.employees SET salary=salary+100 WHERE employee_id=101; 1 row updated.
UPDATE hr.employees SET COMMISSION_PCT=2 WHERE employee_id=101; Session waits enqueue due to lock conflict.	9:00:05 	SELECT sum(salary) FROM hr.employees; SUM(SALARY) ----- 692634
Session still waiting!	16:30:00	Many selects, inserts, updates, and deletes during the last 7.5 hours, but no commits or rollbacks!
1 row updated. Session continues.	16:30:01	commit;

Kemungkinan Penyebab Lock Conflict

- **Perubahan yang tidak dicommit**
- **Transaksi yang berjalan lama**
- **Level tinggi penguncian yang tidak perlu**



Mendeteksi Lock Conflict

Pilih Blocking Session dari Halaman Performance

Database: dba10g > Blocking Sessions

Blocking Sessions

Page Refreshed Nov 25, 2003 6:53:21 PM

View Session Kill Session

Expand All Collapse All

Select	Username	Sessions Blocked	Session ID	Session Serial Number	SQL Hash Value	Wait Class	Wait Event	P1	P2	P3	Seconds in Wait
<input type="radio"/>	Blocking Sessions										
<input checked="" type="radio"/>	HR	1	17	28225		Idle	SQL*Net message from client	1413697536	1	0	44
<input type="radio"/>	HR	0	32	13948	898777808	Application	enq: TX - row lock contention	1415053318	5373958	39	35

Klik link Sesion ID untuk menampilkan informasi tentang locking session, termasuk statment SQL yg sebenarnya

Memecahkan Lock Conflict

Untuk menyelesaikan masalah lock Conflict:

- Buat session yang memegang lock untuk commit atau rollback
- Hentikan Session yang memegang lock hanya jika terpaksa

Session Details: SID 17

Collected From Target: Nov 25, 2003 7:27:19 PM [Refresh](#)


[General](#) [Statistics](#) [Wait Events](#) [Open Cursors](#) [Locks](#)

Hash Value	SQL Text
898777808	update emp set salary=salary+1 where employee_id=100
4204610748	SELECT COUNT(*) FROM ALL_POLICIES V WHERE V.OBJECT_OWNER = :
1522507430	SELECT USER_ID FROM ALL_USERS WHERE USERNAME = :B1

[General](#) [Statistics](#) [Wait Events](#) [Open Cursors](#) [Locks](#)

[Kill Session](#)

Deadlocks

Transaction 1		Transaction 2
<code>UPDATE employees SET salary = salary x 1.1 WHERE employee_id = 1000;</code>	9:00	<code>UPDATE employees SET manager = 1342 WHERE employee_id = 2000;</code>
<code>UPDATE employees SET salary = salary x 1.1 WHERE employee_id = 2000;</code>	9:15	<code>UPDATE employees SET manager = 1342 WHERE employee_id = 1000;</code>
<code>ORA-00060: Deadlock detected while waiting for resource</code>	9:16	

Ringkasan

Pada bab ini, anda seharusnya telah mempelajari bagaimana cara untuk:

- Deteksi dan menyelesaikan masalah lock
- Mengatur deadlock