



墨天轮



中国数据库联盟
All China Database Union

—— DTC 2024 ——

Oracle写冲突解决方案——更新重启

杨廷琨

云和恩墨技术研究院总经理



目录

CONTENTS



01

Oracle的并发策略

02

更新重启的产生

03

更新重启的监控

04

国产化替代的启迪

Oracle的并发策略

● 并发读取

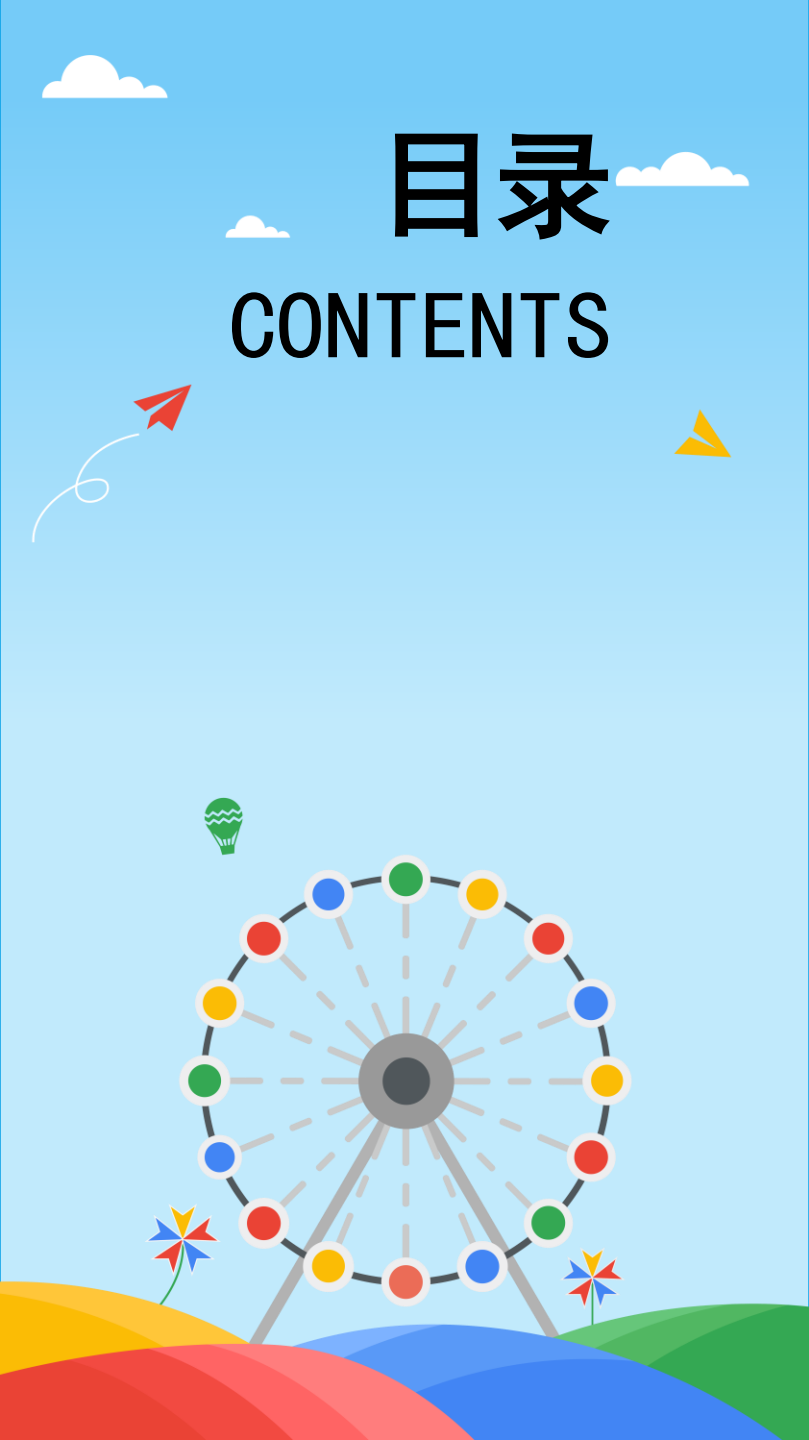
- 读共享
- read by other session

● 并发读写

- 读不阻塞写，写不阻塞读
- 多版本读一致性
- data blocks consistent reads - undo records applied

● 并发修改

- 锁
- Enq: TX – row lock contention
- 更新重启动



目录

CONTENTS

01

Oracle的并发策略

▶ 02

更新重启的产生

03

更新重启的监控

04

国产化替代的启迪

更新重启的产生

● 并发读取

- Oracle读取数据时，会将文件中的块加载到内存中共享。

● 并发读写

- 如果当前读取的数据块被修改过，则会构造读一致性，通过UNDO记录在内存中回滚修改还原数据块的前镜像，把数据块还原到查询发生的时刻
- Oracle实现的读一致性也被称为多版本读一致性，每个会话都会构造自己的一致性查询版本。

● 并发修改

- 对于写操作而言，因为数据只有一份，就只能修改唯一的这份数据
- 读操作，访问的是一致性版本，写操作，修改的永远是当前版本，并发修改相同记录通过加锁来控制一致性
- 写操作执行的过程中，发现要修改的记录发生了变化，破坏了更新发起时刻的一致性，这时就触发了更新重启，也就是说更新操作会放弃之前的修改，然后重新发起一次新的更新操作。

更新重启动的产生

```
SQL> CREATE TABLE T_UPDATE (ID NUMBER);
```

Table created.

```
SQL> INSERT INTO T_UPDATE SELECT ROWNUM FROM DUAL CONNECT BY LEVEL < 4;
```

3 rows created.

```
SQL> COMMIT;
```

Commit complete.

```
SQL> SELECT * FROM T_UPDATE;
```

ID
1
2
3

更新重启动的产生

SESSION 1:	SESSION 2:	SESSION 3:
SQL> SET SQLP 'SQL1> '	SQL> SET SQLP 'SQL2> '	SQL> SET SQLP 'SQL3> '
SQL1> UPDATE T_UPDATE SET ID = 4 WHERE ID > 2; 1 row updated.		
	SQL2> UPDATE T_UPDATE SET ID = ID + 0.1 WHERE ID > 0;	
		SQL3> INSERT INTO T_UPDATE VALUES (5); 1 row created. SQL3> COMMIT; Commit complete.
SQL1> COMMIT; Commit complete.		
	4 rows updated. SQL2> SELECT * FROM T_UPDATE; ID ----- 1.1 2.1 4.1 5.1	

更新重启的产生

● 更新重启

- 更新过程中发现要更新的记录发生修改
- 为了避免更新的结果导致不一致性，已更新记录回滚
- 重新发起更新操作

更新重启动的产生

```
SQL2> ROLLBACK;
```

```
Rollback complete.
```

```
SQL2> DELETE T_UPDATE WHERE ID > 4;
```

```
1 row deleted.
```

```
SQL2> UPDATE T_UPDATE SET ID = 3 WHERE ID = 4;
```

```
1 row updated.
```

```
SQL2> COMMIT;
```

```
Commit complete.
```

```
SQL2> SELECT * FROM T_UPDATE;
```

```
      ID  
-----  
      1  
      2  
      3
```

更新重启的产生

SESSION 1:	SESSION 2:	SESSION 3:
SQL1> UPDATE T_UPDATE SET ID = 4 WHERE ID > 2; 1 row updated.		
	SQL2> UPDATE T_UPDATE SET ID = ID + 0.1;	
		SQL3> INSERT INTO T_UPDATE VALUES (5); 1 row created. SQL3> COMMIT; Commit complete.
SQL1> COMMIT; Commit complete.		
	3 rows updated. SQL2> SELECT * FROM T_UPDATE; ID ----- 1.1 2.1 4.1 5	

更新重启动的产生

```
SQL> CREATE USER U_UPDATE IDENTIFIED BY TESTUPDATE;
```

User created.

```
SQL> GRANT CREATE SESSION TO U_UPDATE;
```

Grant succeeded.

```
SQL> CREATE USER U_OWNER IDENTIFIED BY OWNERTEST;
```

User created.

```
SQL> GRANT CONNECT, RESOURCE TO U_OWNER;
```

Grant succeeded.

```
SQL> GRANT UNLIMITED TABLESPACE TO U_OWNER;
```

Grant succeeded.

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/705323321130011200>