**在线建索引被kill**

云和恩墨(北京)信息技术有限公司

技术顾问 燕鑫

http://www.enmotech.com

**文档控制：**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序** | **版本号** | **更改人** | **日期** | **备注** |
| 1 | 1.0版 | 燕鑫 | 2018-06-20 | 初始版本 |
|  |  |  |  |  |
|  |  |  |  |  |

目录

[1. 发现问题 - 4 -](#_Toc517297652)

[1.1 索引状态为valid，但不可用。 - 4 -](#_Toc517297653)

[1.2 索引重建不了 - 4 -](#_Toc517297654)

[2. 百度问题 - 5 -](#_Toc517297655)

[3. 模拟问题 - 6 -](#_Toc517297656)

[3.1 第一次试验，退出会话 - 6 -](#_Toc517297657)

[3.2 第二次试验，kill进程 - 8 -](#_Toc517297658)

[3.3 又进行了数次试验，与试验一的情况矛盾（不知道为啥） - 9 -](#_Toc517297659)

[4. 处理问题 - 9 -](#_Toc517297660)

[5. 总结 - 10 -](#_Toc517297661)

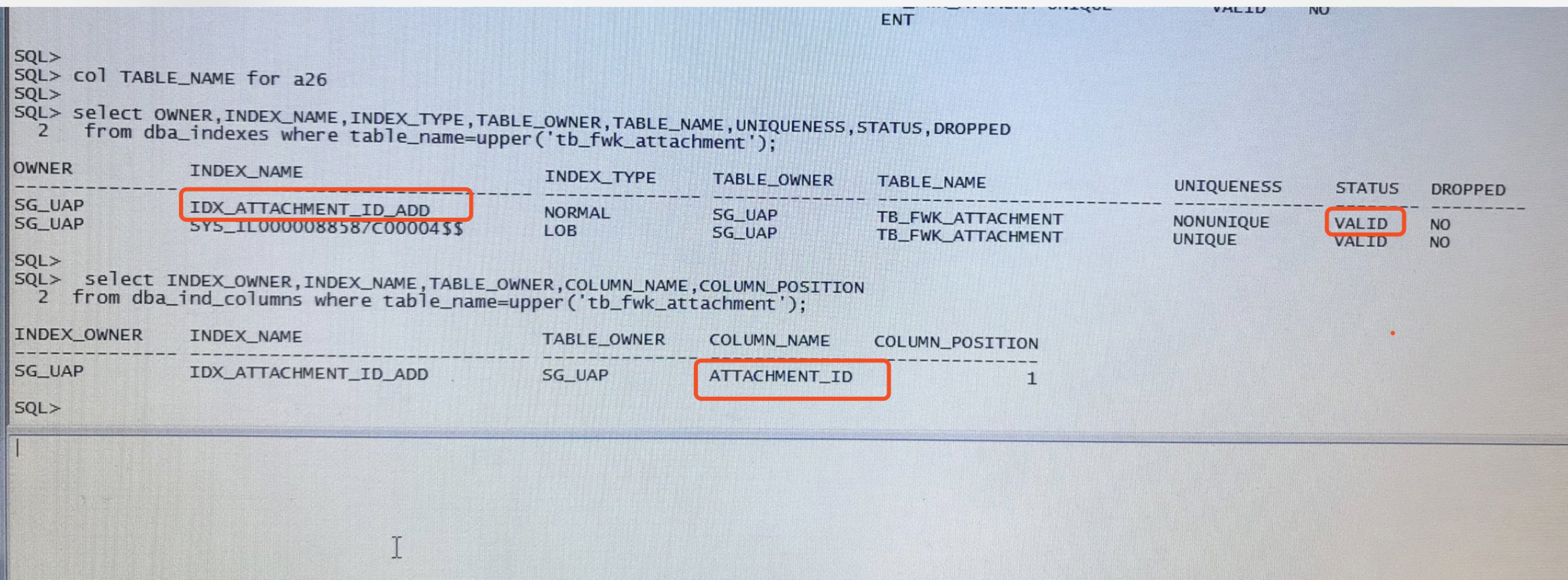
# 发现问题

## 索引状态为valid，但不可用。

该情况是同事李晴晴在工作期间遇到的。

客户通过plsqldeveloper对一张正在进行dml的表在线创建索引，但是由于事务始终没有提交，客户不想等了，就把创建索引的进程给杀了。

然后出现了下面的情况：



发现索引已经创建“成功”，状态为valid。

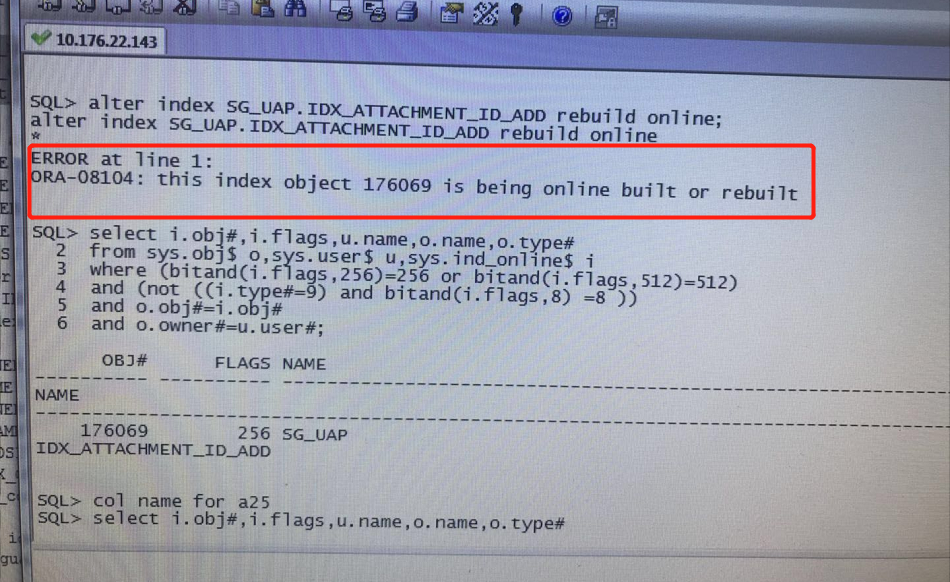
但是当使用索引列搜索，并且用hint强制，都是全表扫描，也就是说，ORACLE是不认这个索引的。

那问题关键就是，为什么这个索引的状态是valid。

## 索引重建不了

删是删不掉，这个主要原因是那条dml还在执行，所以删肯定删不掉，不过后面会做实验，如果dml提交了，其实也删不掉。

当索引再次重建的时候：



可以看到，报了08104的错误。

# 百度问题

经过百度，得到了解决方案：

<https://blog.csdn.net/haibusuanyun/article/details/50236057>

最终采用的就是这里面的第一种解决方案：

declare

isClean boolean;

begin

isClean := FALSE;

while isClean=FALSE loop

isClean := dbms\_repair.online\_index\_clean(dbms\_repair.all\_index\_id,dbms\_repair.lock\_wait);

dbms\_lock.sleep(2);

end loop;

exception

when others then

RAISE;

end;

/

但是我们是指定object\_id，如果按他那样，会扫描所有的处于online built的索引。

这个没有做明确的试验，因为生产库上不敢做，mos上说会扫描所有的index，会非常耗时，尽管这篇mos有点儿问题，但是还是不敢做这个尝试。我之所以推断只会扫描处于online built状态的索引，是因为，如果你给这个函数一个正常的索引的objectid，该函数会报错。

参考文档：**1378173.1**

ORA-24120 covers the following:

a negative (or NULL) index object\_id was specified

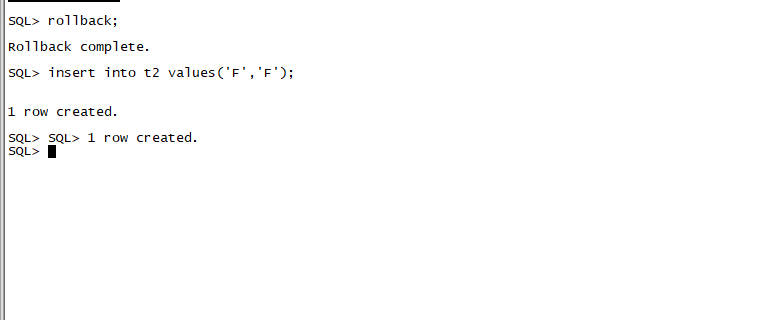
a negative (or NULL) wait\_for\_lock was specified

the given object\_id was not an index object that needed cleanup

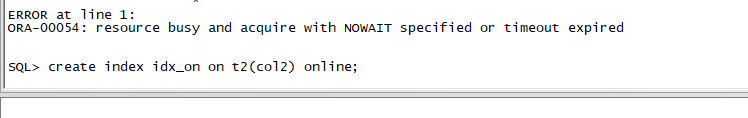
# 模拟问题

## 第一次试验，退出会话

开俩会话：一个会话先发起一个事务：

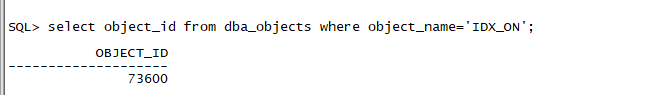


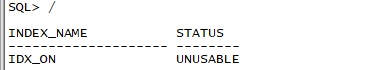
另一个会话进行index创建：



显然，不加onine会报资源忙，加了online以后，创建索引的行为进入等待状态，等待事务完成。

这时候我们查视图：

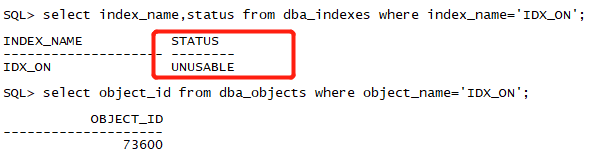




但我们发现这时候索引的状态是unusable的，这是可以理解的，那我们把会话杀掉，看看会不会变成valid。

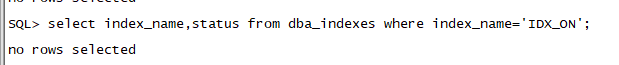
这是我第一次试验，我没有选择杀掉会话进程，因为我觉得直接退出会话效果应该是一样的（因为直接ctrl+c不行）。

退出后再查：



发现这条记录并没有消失，但状态依旧是unusable。

这是我把事务进行了提交。再查：

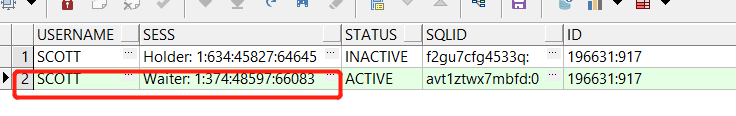


记录消失了，这次是对的状况，那跟客户那边的情况哪里不一样呢？就是客户是把进程kill了，而我是直接退出的会话。

## 第二次试验，kill进程

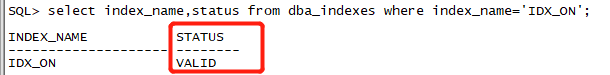
环境模拟还是同前，一个会话进行事务，然后再开一个会话在线建索引。

然后我们通过block脚本，查出如下结果：



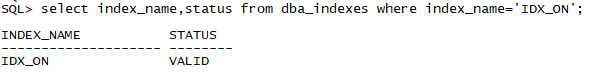
显然这个waiter就是建索引的行为。然后我把它kill了。

然后对索引进行查询：



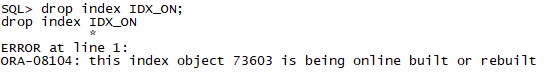
显然，这次模拟是成功的。

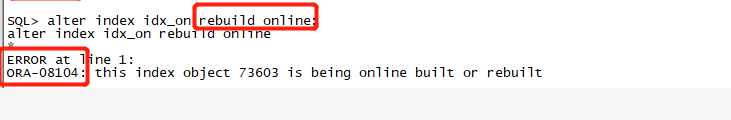
那么我们把事务提交了，这条记录会消失么？



并没有消失！

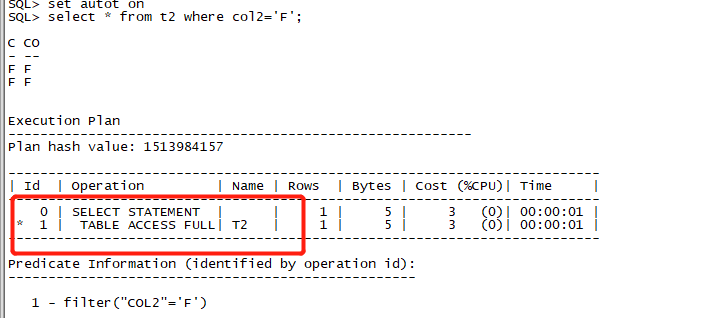
现在我们来删索引或者重建索引：





都不行！

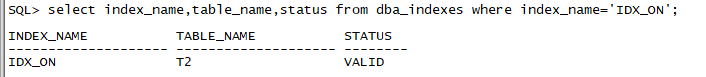
我们来看看这个索引到底能不能用到：



这些李晴晴在生产环境都试过了，所以我也就不试hint强制走索引了。

## 又进行了数次试验，与试验一的情况矛盾（不知道为啥）

这些试验中，每一次在create index online的等待过程中，我查看索引的状态都是valid：



也就是说，status是不是valid跟kill会话没关系。只要用online的方式建索引，那状态就是valid。这里真的不懂为什么同样的试验条件会出现俩种不同的情况。

而且我之后不kill会话，直接退出会话，提交事务，有的时候，index是真的能用，有的时候就是我们这次遇到的问题这样。

但是kill会话，是一定会导致该问题发生的。

# 处理问题

按照之前的链接，稍作了一些修改：

declare

a boolean;

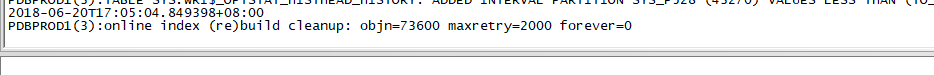
begin

a := dbms\_repair.online\_index\_clean(73603,dbms\_repair.lock\_wait);

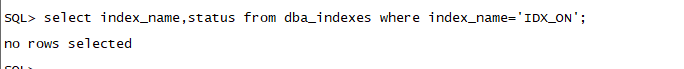
end;

/

我们观察日志有如下记录（虽然不懂啥意思）：



存储过程成功执行后，我们再查视图：



记录消失了，问题解决。

注意：在事务还没完成的情况下，该清理行为会保持等待状态。

# 总结

1. 其实这次是先百度出来，才让李晴晴在生产环境上尝试rebuild online。之前没想到会有这种情况所以，没做任何操作，希望保留现场，但也忘了百度，第二天才反映过来。所以还是得遇事不决先百度。
2. 开始不是没有想到试验，而是思路局限在了怎么构造一张大表来有足够的时间让我kill会话，但是却忽略了真正导致建索引行为等待的是未提交的事务。
3. 总的来说这种情况比较少见，但是我们在迁移前会进行kill local=no的进程，有可能导致这种情况发生。