



高性能DDL技巧

2019/10/17

刘永甫

18620308065

oracle_tigerliu@qq.com

业务微信: ora_service

大表复制方法-1

- CTAS大表复制

- create table new_big **parallel *n* nologging** as select * from source_tab;
- alter table new_big **noparallel**;
- 是否选择nologging要根据需要，含lob字段的表，12c之前不支持并行
- 字段上的**default**和**constraint**设置**不会复制**（not null属性会跟随复制）；
- 复制完，同样用并行增加主键和相关索引

大表复制方法-2

- IAS 复制方法

- 先创建空表（用原表完整DDL，带default 设置等，不要带主键索引）

- 再insert /*+ parallel(8) enable_parallel_dml */ into new_big select * from source_tab;

- 注：12c+才支持enable_parallel_dml ;

- 11g及以下版本步骤要复杂一些：

- alter session enable parallel dml;

- insert /*+ parallel(8) */ into new_big select * from source_tab;

- commit;

- alter session disable parallel dml;

- 然后再增加主键和相关索引

大表move 或 partition move, 启用并行

- 碎片整理, 降低高水位, 转移表空间
 - alter table <tname> move **parallel n** [tablespace tbs_name];
 - 表上所有索引都需要rebuild
 - 同样需要使用parallel: alter index idx_name rebuild [online] **parallel n** [tablespace tbs_name];
 - 注意: lob 字段 (Out-of-Line storage 类型) 不随表一起move, 需要单独move
 - alter table <tname> move lob(<cname>) store as (tablespace <tbs_name>) **parallel n**;
- 精细化管理, 将历史分区转移到普通存储所在表空间
 - 表分区: alter table <tname> move partition <pname> [tablespace tbs_name] **parallel n**;
 - 同样需要rebuild 相关索引 (包括local index和global index)
 - 表分区上的lob字段: alter table <tname> move partition <pname> lob (<cname>) store as (tablespace <tablespace_name>) **parallel n**;
- 注意:
 - lob字段的move 是否支持parallel, 各版本可能有不同, 建议实测!

DISABLE
STORAGE IN ROW

大表增加主键:

- 1、创建唯一索引 `nologging` 可选（根据实际情况），关键是要使用并行

```
create unique index &pk_name on &tab_name(col_name) parallel 8 nologging  
tablespace &tbs_name;
```

- 2、增加约束:

```
alter table &tab_name add constraint &pk_name primary key (col_name) using  
index &pk_name enable novalidate;
```

- 3、别忘了把并行度属性去掉:

```
alter index &idx_name noparallel;
```

大分区表创建索引-1

1. 先创建 **unusable** invisible local index (因为rebuild过程没有global统计信息, 先invisible) 这一步很快, 基本上不需要时间:

```
create index &idx_name on &tab_name(col1,col2 )  
local unusable invisible tablespace (&tbs_name);
```

2. 逐个分区rebuild local index(适当加一点并行), online 和 nologging, 根据需要使用:

```
select 'alter index ' || c.index_name || ' rebuild partition ' || c.partition_name || '  
parallel(degree 4) online nologging;  
from dba_ind_partitions c where index_name=upper('&idx_name') and status != 'USABLE';
```

上面sql生成的结果, 逐个执行! 也可以在多个session同时执行, 并发数量不宜太多

大分区表创建索引-2

3.收集索引上的global 统计信息(重要!):

exec

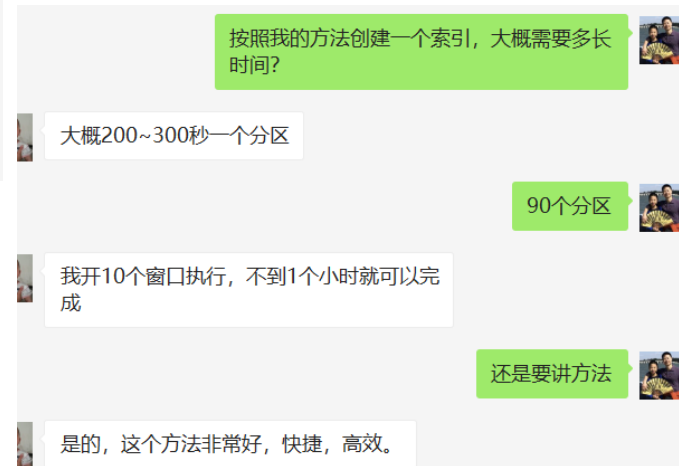
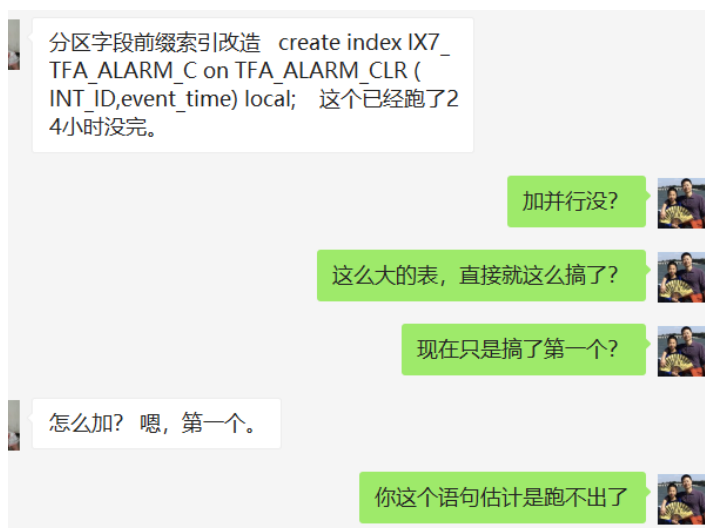
```
dbms_stats.gather_index_stats('&owner','&idx_name',degree=>8 ,force=>true);
```

4. 去掉索引并行属性:

```
alter index &idx_name noparallel;
```

5. 使索引可见:

```
alter index &idx_name visible;
```



大表快速增加字段

- Add column--**快速**增加字段:

10g:不要设置默认值
11g:可以设置默认值, 需要not null
12c:可以设置默认值, 也可以null

- 10g : alter table t_big add new_col number ;

- 如果加了default, 会对当前所有记录逐条做update, 比较慢

- 11g : alter table t_big add new_col number **default 0 not null**;

- 如果是可为null, 也会对当前所有记录逐条做update, 比较慢

- 12c : alter table t_big add new_col number **default 0**;

- 有default值, 也可以为null

优化，让数据库高效稳定运行