RE: Oracle cursor help

Source: http://coding.derkeiler.com/Archive/Perl/perl.dbi.users/2005−06/msg00171.html

- *From*: Ron.Reidy@xxxxxxxxxxxxxxxxx (Ron Reidy)
- *Date*: Thu, 23 Jun 2005 10:49:58 −0600

Row−at−a−time processing is horrifically slow. If there is a large amount (gigs) of data, he should use bulk processing in PL/SQL, possibly using the LIMIT clause.

Something like this (your mileage may vary):

```
declare
rids dbms_utility.uncl_array;
begin
select rowidtochar(rowid)
from table
where <your where clause>
bulk collect into rids;

for all idx in rids.first .. rids.last
delete from table where rowid = chartorowid(rids(indx));
end;
/

commit;
```

No need to lock the rows.

If there are more rows to delete than to keep, he should consider making a copy of the data (using CTAS) into a temporary table and then:

1. drop all indexes
2. truncate the table
3. drop the table
4. rename temp table to original_table_name

−−−−−−−−−−−−−−−−−
Ron Reidy
Lead DBA
Array BioPharma, Inc.

−−−−−Original Message−−−−−
From: Job Miller [mailto:jobmiller@xxxxxxxxx]
Sent: Thursday, June 23, 2005 10:41 AM

To: jseger3@xxxxxxxxxxxx; dbi−users@xxxxxxxx
Subject: Re: Oracle cursor help

Size your rollback so you can do this in one
transaction.

DECLARE

CURSOR T1 IS
SELECT e, f
FROM T1
WHERE e < f
FOR UPDATE;
BEGIN
OPEN T1Cursor;
LOOP

/* Retrieve each row of the result of the
above query

into PL/SQL variables: */

12) FETCH T1Cursor INTO a, b;

/* If there are no more rows to fetch,
exit the loop: */

13) EXIT WHEN T1Cursor%NOTFOUND;

/* Delete the current tuple: */

14) DELETE FROM T1 WHERE CURRENT OF T1Cursor;

/* Insert the reverse tuple: */

15) INSERT INTO T1 VALUES(b, a);

16) END LOOP;

/* Free cursor used by the query. */

17) CLOSE T1Cursor;

18) END;

−−− jeff <jseger3@xxxxxxxxxxxx> wrote:

> ditto....and I'd add that if your delete is taking
> forever to run that
> you may want to consider getting some indexes on the

> columns you are
> using to filter for your delete. If a "select *
> from myTable where
> <insert your predicate here>" takes a long time,
> then so will a delete
> with the same predicate and no PL/SQL trick is going
> to speed it up. You
> may be able to thow in a hint, depending on how the
> table is
> structured/indexed, and it's also possible that
> running stats on the
> table wil have a positive effect on the speed.
>
> Martin Hall wrote:
>
> > Well, I have to echo what others have said. The
> only reason I can
> > think of for using loops to do your delete is if
> you really want to
> > save a chunk of rollback segment. Even then, I
> would consider
> > batching the rows up into chunks that you could
> manage with predicates
> > like BETWEEN or IN. Switching context from PL/SQL
> for the loop to SQL
> > for the delete takes time (only a little, but it
> adds up). And if
> > you're using bitmap indexes, batch deletes are
> much kinder than single
> > row ones.
> >
> > Martin (apologies if this is OT)
> >
> > Vergara, Michael (TEM) wrote:
> >
> >> Steve:
> >>
> >> I have a PL/SQL process that I use to purge rows
> from a table
> >> that grows by about 14M rows/month. Even though
> it's PL/SQL,
> >> the process should be (might be?) transportable
> to Perl.
> >>
> >> Basically, it looks like this:
> >>
> >> Define SQL statement to gather the rows (PL/SQL
> cursor);
> >> ctr = 0;
> >> For each row:
> >> delete from table where recid = cursor.recid;

> >> ctr += 1;
> >> if ctr > 10000 {
> >> commit;
> >> ctr = 0;
> >> }
> >> next
> >>
> >> I don't know enough about the DBI to know if the
> 'recid'
> >> field from PL/SQL can be used in Perl, but if you
> get it
> >> to work I'd be interested in hearing about it.
> >>
> >> Cheers,
> >> Mike
> >>
> >>
> >> −−−−−Original Message−−−−−
> >> From: Steve Sapovits
> [mailto:steves06@xxxxxxxxxxx] Sent: Wednesday,
> >> June 22, 2005 11:18 PM
> >> To: dbi−users@xxxxxxxx
> >> Subject: Oracle cursor help
> >>
> >>
> >> I have a few Oracle tables that have gotten way
> too big. Normal
> >> DELETE/WHERE clauses consume too many resources.
> I know enough
> >> to know that cursors would be more efficient here
> but I don't
> >> know enough to actually start using them. Does
> anyone have an
> >> example or two (or three) to bootstrap me? What
> I'd essentially
> >> want to do is loop through all rows, look at a
> few column settings,
> >> and delete those that fall into certain
> categories (e.g., certain
> >> dates being so old and certain status fields not
> being set to a
> >> set value, etc.). I can also limit the row set
> up front with a
> >> pretty simple WHERE clause −− whatever is more
> efficient is what
> >> I'm aiming for (it wasn't clear to me if limiting
> up front with
> >> WHERE clauses defeated the efficiency of cursors
> ...)
> >>
> >>

> >>
> >
> >
> >
>
>

.

- Prev by Date: ***Re: Oracle cursor help***
- Next by Date: ***NOT getting "duplicate entry" error with DBI/MySQL***
- Previous by thread: ***Re: Oracle cursor help***
- Next by thread: ***RE: Oracle cursor help***
- Index(es):
  - ***Date***
  - ***Thread***