

Re: Locks, Scope & Performance

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.programming/2005-02/3862.html>

From: Louis Davidson (*dr_dontspamme_sql_at_hotmail.com*)

Date: 02/15/05

Date: Mon, 14 Feb 2005 22:50:35 -0500

Understand that NOLOCK can give you invalid results, since it can see dirty data. If having consistent results are critical, this is not a good idea.

If you can live with possible spurious results, based on rows being picked up that turn out to have been invalid, then it is fine. I don't know about your architecture, but chances are you will have very little of this unless you have quite high throughput, but it is a concern.

A million rows is quite a few, but it is not an incredible amount. Consider optimizing your query, and you might only have blocks that last a second or two if any. A query that brings the server to its knees is often a sign that you have table scans that are slowly walking through your table. It might also be taking a table lock rather than rowlocks based on resource utilization. This query could be the problem, but it could also be your update queries or others. Can you tell us which query is blocking which?

--

Louis Davidson - drsql@hotmail.com

SQL Server MVP

Compass Technology Management - www.compass.net

Pro SQL Server 2000 Database Design -

<http://www.apress.com/book/bookDisplay.html?bID=266>

Blog - <http://spaces.msn.com/members/drsql/>

Note: Please reply to the newsgroups only unless you are interested in consulting services. All other replies may be ignored :)

"JDP@Work" <JPGMTNoSpam@sbcglobal.net> wrote in message

news:ezFTA9uEFHA.3200@TK2MSFTNGP10.phx.gbl...

> Thanks to pdxJaxon, I'm looking into with NOLOCK

>

> if my exec(@str) from looks like...

>

> from

> '+@db+'.dbo.CONTACT1 contact1

> ,'+@db+'.dbo.CONTACT2 contact2

> ,'+@db+'.dbo.CAL cal

> ,'+@db+'.dbo.conthist ch

> ,'+@db+'.dbo.users

> where

>

> ..adding...

> from

> '+@db+'.dbo.CONTACT1 contact1 with NOLOCK

microsoft.public.sqlserver.programming: Re: Locks, Scope & Performance

```
> ,'+@db+'.dbo.CONTACT2 contact2 with NOLOCK
> ,'+@db+'.dbo.CAL cal with NOLOCK
> ,'+@db+'.dbo.conthist ch with NOLOCK
> ,'+@db+'.dbo.users with NOLOCK
> where
>
> Lee,
>
> I moved away from using the temp db #mytable to creating a table
> dynamically
> named as noted in my pseudo snipit.
>
> I did this because it improved performance from some now in place
> processes that
> used to bring the server to it's knees.
>
> However I'm still interested and would appreciate a clarification
>
>> e.g. temp table which can be filled within the sub procedure and then the
>> calling procedure would get recompiled if necessary for the next step.
>
> 1. Is there any difference between using a table dynamically named within
> my
> outer sProc and #myTemp?
>
> 2. Would the process you are suggesting look like a proc that calls a sub
> proc
> and creates w/recompile an inner sProc?
>
> create proc my_sProc_Main_Outer
>
> exec mysProc_sub inserting rows to a (temp) table
>
> then...
>
> create proc my_sProc_Main_Inner
>
> with recompile
>
> as
> .....
> return
>
> drop table(s)
>
> return
>
> Exec'ing the sProc for the report would look something like.....
>
> exec my_sProc_Main_Outer @db ,@w_str ,@action
>
> TIA
>
> JeffP....
>
>
> "Lee Tudor" <mr_tea@ntlworld.com> wrote in message
> news:A5aQd.547$Se3.306@newsfe5-win.ntli.net...
>> SELECT still issues shared resource locks and INSERT EXEC runs the entire
>> sub
>> procedure within the INSERT transaction. This combined with the fact you
>> are
```

microsoft.public.sqlserver.programming: Re: Locks, Scope & Performance

```
>> selecting from 1M rows is most likely your culprit.
>>
>> You could override the locking policy on certian tables and tell it to
>> use
>> page locks or no locks instead. or change the transaction isolation level
>> to
>> a more lax level.
>>
>> You may want to consider a different way of storing the intermediate
>> data,
>> e.g. temp table which can be filled within the sub procedure and then the
>> calling procedure would get recompiled if necessary for the next step.
>>
>> This would address the cause rather than the symptom and probably give a
>> significant performance boost into the bargain.
>>
>> Mr Tea
>>
>> "JDP@Work" <JPGMTNoSpam@sbcglobal.net> wrote in message
>> news:%23wEsvHuEFHA.628@TK2MSFTNGP15.phx.gbl...
>> > I'm getting LOCKS caused by a particular report that uses two sProcs to
>> > render a
>> > report.
>> >
>> > I'm confused as to why the following should be locking my contact
>> > calendar
>> > and
>> > history tables, they just do a select into a table, shouldn't this be
>> > read
>> > only
>> > and not cause any locks?
>> >
>> > Is there a way to be sure that I exec as read only and do not create a
>> > table
>> > lock?
>> >
>> > Details.....
>> >
>> > The 1st sProc is called (nested) from within the second as
>> >
>> > create proc my_sProc_Main
>> >
>> > declare some vars, @tmpTbl varchar(36)
>> >
>> > set @tmpTbl = replace('-',newID(),'9')
>> >
>> > create a real table..... exec(create table table tblWorkingName'+@
>> > tmpTbl)
>> >
>> > Insert myTable exec my_sProc_Sub
>> > (my_sProc_Sub - select complex conditions, union another set of complex
>> > conditions, scan 1m row table return 1200 rows)
>> >
>> > create another real results table..... exec(create table
>> > tblResultsName'+@
>> > tmpTbl)
>> >
>> >
>> > Evaluate data in the first table (containing 1200 rows) and insert into
>> > the
>> > second table about 20 resulting rows
>> >
```

microsoft.public.sqlserver.programming: Re: Locks, Scope & Performance

```
>> > return the results of the second table
>> >
>> > drop both tables
>> >
>> > Total process time between 13 and 56 seconds
>> >
>> > I run similar processes, some as complex but not using two sProcs,
>> > however
>> > this
>> > one report brings my server to its knees.
>> >
>> > The production server is a Compaq dual 700ghz, 1gb Ram
>> >
>> > My dev box is a single 1.4ghz, 1gb Ram and it execs somewhat faster and
>> > with
>> > half the server killing effects.
>> >
>> > TIA
>> >
>> > JeffP.....
>> >
>> >
>>
>
>
```