

microsoft.public.sqlserver.server: Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.server/2004-11/2774.html>

From: Andrew J. Kelly (sqlmvpnoospam_at_shadhawk.com)

Date: 11/20/04

Date: Sat, 20 Nov 2004 12:16:35 -0500

Running a query or sp in Query Analyzer is not always the same as running it from your application. Maybe this can shed some light on your issue:

<http://tinyurl.com/159s> Parameter sniffing

```
--
Andrew J. Kelly  SQL MVP
"ktf" <ktf@discussions.microsoft.com> wrote in message
news:AEB4BA06-A40A-418E-B889-181BB85319A0@microsoft.com...
> Steve,
> Thank you very much for your very helpful explanation. I really rebuild
> all
> indexes in the order which I received through the patch and that was the
> last
> time. For the first time after six month I finally got the approval to
> rebuild indexes. I ran a dynamic db reindex to minimize human errors and
> establish a proper procedure document for the future DBAs for this complex
> project. So after that I was facing this high I/O reads. After two weeks I
> finally got the approval to recreate all index through running hard coded
> script, which I extracted from the database then I worked over night to
> put
> them in order. Before I was using this method (hard coded) and had done
> that
> many times. I was actually using the migration time to recreate indexes.
> This
> is a 24/7 mission critical system.
> So I even try to recreate them by orders.
>
> Andrew,
> Thank you for your answer. Tonight I created additional indexes just to
> match the exact where clauses but it did not do it.
>
> The bottom line is I see index scans all over the places but not seeks. No
> the trace of the query analyzer does not show a high number but the trace
> I
> put out there shows three million vs under 100. The same trace was showing
> very low number before I do the db reindex. I do not know if this is a
> security issue or the standard method caused this problem.
>
> Is there a place or function to see the mode of indexes or something like
> disable or enable index seeks? If there is how do I find how or who made
> that
```

Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

microsoft.public.sqlserver.server: Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

```
> change. How should turn that back? Have you experienced this situation? I
> am
> really running out of time.
> Thank you
>
> "Steve Kass" wrote:
>
>> ktf,
>>
>> You mention an index scan in the execution plan. Is it
>> possible that the order of that index is such that the rows
>> you need to find are not going to be found anywhere near
>> the beginning? (Equivalently, does that scan show a small
>> number for estimated rowcount, yet a huge actual
>> rowcount if you look at the actual plan?)
>>
>> When I've seen dramatic differences between the
>> estimated execution plan estimated I/O and the actual
>> I/O, it has usually been due to correlated data that
>> undermine the query optimizer's cost estimation. It's
>> particularly bad when the extra I/O involves bookmark
>> lookups. (The solution is usually to add an index hint, change
>> indexing to avoid needing bookmark operations, or rewrite
>> a search predicate so that it is not a SARG.)
>>
>> It would really help to see more detail here, but I'll give a
>> quick example of what could be going on.
>>
>> --Scenario
>> The query is something like "find the oldest unpaid invoice".
>> There is a nonclustered index on (invoiceDate).
>> The Invoices table has 10,000,000 rows, 300,000 of them "unpaid".
>> All the unpaid invoices are among the most recent 1,000,000 invoices.
>>
>> This is a disaster. The optimizer knows that 3%
>> of the table has the value "unpaid", and assumes that
>> any way it searches, it will only have to look through about
>> 33 rows to find "unpaid". So why not search in invoiceDate
>> order, so the first "unpaid" row will be the one you want?
>> That's much better than finding all 300,000 "unpaid" invoices
>> (assuming that column is also indexed), and then choosing the
>> earliest of those.
>>
>> Even if the invoiceDate index doesn't contain the paid/unpaid
>> column, you can go grab that value from the clustered index
>> 30 or 35 times without any trouble.
>>
>> Unfortunately, instead of finding "unpaid" within the first
>> 33 or so rows inspected, "unpaid" is not found until
>> more than 9,000,000 rows are inspected, each having
>> required a bookmark lookup in the clustered index.
>> The optimizer's estimation was performed correctly,
>> but the correlated data meant the estimate was very
>> far off.
>>
>> Steve Kass
>> Drew University
>>
>>
>> ktf wrote:
>>
>> >Mr. Kelly,
```

Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

microsoft.public.sqlserver.server: Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

```
>> >I knew the definition for the DDL but for security reason I cannot
>> >provide
>> >the real col names. I needed to get general Idea for the root of my
>> >problems.
>> >Today I recreated indexes for the use of that sp on a different device
>> >with
>> >different names followed by update statistics and sp_compile sp name. It
>> >did
>> >not help me. Is there any tools I can use somewhere between the
>> >execution
>> >plan and the trace that can help me where the problem is. I understood
>> >when I
>> >recreate indexes the data within is not changing. How do I pinpoint the
>> >problem within the index?
>> >v/r
>> >ktf
>> >
>> >
>> >"Andrew J. Kelly" wrote:
>> >
>> >
>> >
>> >>Without seeing the DDL I can't say. DDL stands for Data Definition
>> >>Language. When someone asks for the DDL they want to see how the table
>> >>is
>> >>actually defined. You can script the table and indexes with Enterprise
>> >>Manager and post that here so we can see what we are up against.
>> >>
>> >>--
>> >>Andrew J. Kelly SQL MVP
>> >>
>> >>
>> >>"ktf" <ktf@discussions.microsoft.com> wrote in message
>> >>news:04E87FB9-4F77-4A10-874B-2DB7284EDC2B@microsoft.com...
>> >>
>> >>
>> >>>I did not get about: Can you post the DDL for the table and the
>> >>>indexes and
>> >>>the query that is causing all these reads?
>> >>>I see Index Scan on the execution plan for that case what I should do?
>> >>>Thank you,
>> >>>ktf
>> >>>
>> >>>"Andrew J. Kelly" wrote:
>> >>>
>> >>>
>> >>>
>> >>>>First off there is no need to do an sp_recompile if you just updated
>> >>>>the
>> >>>>stats. Anytime the stats get updated it will force any plan that
>> >>>>uses
>> >>>>those
>> >>>>stats to be recompiled the next time they are run. If you used DBCC
>> >>>>DBREINDEX to reindex the tables then it also updates the stats and
>> >>>>the
>> >>>>manual updating of them is redundant. If the number of reads went
>> >>>>from
>> >>>>under
>> >>>>100 to over 3 million you are most likely doing a table or index
>> >>>>scan.
>> >>>>Can
>> >>>>you post the DDL for the table and the indexes and the query that is
```

Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

microsoft.public.sqlserver.server: Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

```
>> >>>>causing
>> >>>>all these reads? Also how many rows are there that match that value
>> >>>>and
>> >>>>how
>> >>>>many total?
>> >>>>
>> >>>>--
>> >>>>Andrew J. Kelly SQL MVP
>> >>>>
>> >>>>
>> >>>>"ktf" <ktf@discussions.microsoft.com> wrote in message
>> >>>>news:F223DE8B-90DD-433F-8537-574FFA530290@microsoft.com...
>> >>>>
>> >>>>
>> >>>>>Mr. Kelly and Mr. Hodgson,
>> >>>>>I did bounce between my active passive nodes before I go live on the
>> >>>>>production. For a particular stored procedure I had very few number
>> >>>>>of
>> >>>>>reads
>> >>>>>but after reindexing/ update statistics/ shrinking by file the
>> >>>>>number
>> >>>>>of
>> >>>>>reads for the same thing went up very high.
>> >>>>>Today I recreated indexes for the use of that only stored procedure,
>> >>>>>ran
>> >>>>>update statistics on the those tables involved then did the
>> >>>>>sp_recomplile
>> >>>>>stored procedure name. I ran the trace but the number is still way
>> >>>>>up 3
>> >>>>>Million vs under 100.
>> >>>>>I even used the Index Tuning Wizard, did the update statistics the
>> >>>>>sp_recompile, and still did not help me.
>> >>>>>
>> >>>>>The question I have is if I add a new index file device and create
>> >>>>>those
>> >>>>>indexes with drop existing option then drop the previous (old) index
>> >>>>>file,
>> >>>>>will it help me.
>> >>>>>
>> >>>>>All I want to do is to force the sql server to use indexes. How do I
>> >>>>>do
>> >>>>>that
>> >>>>>other than what I had done. I need your help asap.
>> >>>>>Thank you,
>> >>>>>ktf
>> >>>>>
>> >>>>>"Andrew J. Kelly" wrote:
>> >>>>>
>> >>>>>
>> >>>>>
>> >>>>>>If you reindexed and then did a shrink you just destroyed all the
>> >>>>>>good
>> >>>>>>work
>> >>>>>>you did in the reindex. A shrink operation will move pages from
>> >>>>>>the
>> >>>>>>rear
>> >>>>>>of
>> >>>>>>the file to the front and will most definately fragment your tables
>> >>>>>>and
>> >>>>>>indexes all over again. A shrink is usually a bad idea. The db
>> >>>>>>needs
>> >>>>>>a
```

Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

microsoft.public.sqlserver.server: Re: Why I/Oread is a low Num on the exec. Plan but very high on tr

>> >>
>>