

Re: Poor performance after upgrading to sql server 2005

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.setup/2006-03/msg00029.html>

- *From:* Rich Wood <RichWood@xxxxxxxxxxxxxxxxxxxx>
 - *Date:* Thu, 2 Mar 2006 11:25:15 -0800
-

There are definitely a lot of references to parallelism in the execution plan on the SQL 2005 machine. I "disabled" the other processors by unchecking the Processor Affinity and I/O Affinity check boxes in the Server Properties gui for all but the first processor. The result was no difference at all in performance.

When I look in the Memory tab of the Server Properties gui it shows 8192 MB and 10240 MB as the minimum and maximum memory values respectively. Interesting that it's different than what sp_configure reports.

It seems to me that what's happening is that, instead of evaluating the sub-query once which is all that is necessary, it gets evaluated for every record of the outer query. The question is, why is it treating the sub-query differently in SQL 2005?

One other piece of evidence... we have a subquery that looks like this:

```
AND 1 = CASE WHEN @FREQUENCY = 'W'  
THEN ( SELECT 1 WHERE CURR.DATA_ID IN (SELECT DATA_ID FROM fdINDDATA_DAILY  
D (NOLOCK)))  
ELSE ( SELECT 1)  
END
```

With this syntax the query runs really, really slow. However, if I replace the @FREQUENCY variable with a string, say 'D' for example, it runs really fast. Again, it appears that when I use the variable, it evaluates the sub-query for each outer record, but it does not when I hard code a string.

Thanks,

Rich Wood

"Andrew J. Kelly" wrote:

Re: Poor performance after upgrading to sql server 2005

See in-line:

--

Andrew J. Kelly SQL MVP

"Rich Wood" <RichWood@xxxxxxxxxxxxxxxxxxxx> wrote in message
news:95AC41A7-D52D-4463-BA82-73E81903710E@xxxxxxxxxxxxxxxxxxxx

The databases are exactly the same, so the number of rows generated by the two queries are also exactly the same.

What I was getting at is if the plans are not identical then you can get joins in different orders or with different tables. This can dramatically affect the number of rows touched in each step even if the final outcome is the same. If on one plan it joined TableA & B then C where as the other was TableB & C then A it can matter quite a bit. Or even the type of join or operation is a factor.

As far as the HEAP, the indexes that were fragmented which I tried to rebuild are all clustered indexes, not HEAPs, as reported in the dm_db_index_physical_stats table. Since this is an exact copy of a sql 2000 database with the same indexes etc., after optimizing the indexes and re-generating the statistics, I don't understand why a query with a sub-query would run so much slower in sql 2005. Unless sql 2005 treats sub-queries fundamentally differently.

You mentioned the fragmentation didn't change so I wanted to be sure you were not using Heaps. Unless the fragmentation is trivial I still wonder why there was no change.

RE setting the MAXDOP, this is from Books OnLine:

Parallel index operations are available only in SQL Server 2005 Enterprise Edition.

That is referring to things like Create Index, ALTER INDEX etc. Parallel queries are still very much alive.

Re: Poor performance after upgrading to sql server 2005

What can I do with MAXDOP in sql 2005 standard edition? Looking at perfmon output I can see that multiple CPUs are being used. I have tried disabling all but one CPU but without any performance differences in the tests I've run.

Perfmon does not tell you how they are being used. You need to look at the query plan to see if it used paralelism or not. There are times when it can help and times when it can hurt. How were you disabling it?

As for the sp_configure below I see you have changed a few settings from the default.

You have Min and Max memory set to roughly 2GB. Why? You have 12GB on the machine why are you limiting it to only 2GB? Why set a min at all?

Here are the advanced options:

```
affinity mask -2147483648 2147483647 0 0
allow updates 0 1 0 0
awe enabled 0 1 0 0
c2 audit mode 0 1 0 0
cost threshold for parallelism 0 32767 5 5
Cross DB Ownership Chaining 0 1 0 0
cursor threshold -1 2147483647 -1 -1
default full-text language 0 2147483647 1033 1033
default language 0 9999 0 0
fill factor (%) 0 100 0 0
index create memory (KB) 704 2147483647 0 0
lightweight pooling 0 1 0 0
locks 5000 2147483647 0 0
max degree of parallelism 0 32 0 0
max server memory (MB) 4 2147483647 2023 2023
max text repl size (B) 0 2147483647 65536 65536
max worker threads 32 32767 255 255
media retention 0 365 0 0
min memory per query (KB) 512 2147483647 1024 1024
min server memory (MB) 0 2147483647 2023 2023
nested triggers 0 1 1 1
network packet size (B) 512 65536 4096 4096
open objects 0 2147483647 0 0
priority boost 0 1 0 0
query governor cost limit 0 2147483647 0 0
query wait (s) -1 2147483647 -1 -1
recovery interval (min) 0 32767 0 0
remote access 0 1 1 1
remote login timeout (s) 0 2147483647 20 20
```

Re: Poor performance after upgrading to sql server 2005

```
remote proc trans 0 1 0 0
remote query timeout (s) 0 2147483647 600 600
scan for startup procs 0 1 0 0
set working set size 0 1 0 0
show advanced options 0 1 1 1
two digit year cutoff 1753 9999 2049 2049
user connections 0 32767 0 0
user options 0 32767 0 0
```

Thanks again,

Rich Wood

"Andrew J. Kelly" wrote:

If they are not exactly the same as in the types of operations and the order etc it can make a big difference in the CPU for a given step. Especially if the number of rows at that step is different. The costs are calculated differently between the two versions so you can't compare costs. You might want to SET STATISTICS IO ON before each query and compare those to see what may be different.

Standard Edition still honors MAXDOP as long as you have more than 1 CPU for queries and such. It is just things like Reindexing and DBCC's that will not use parallelism in Std edition. As for rebuilding the index I am willing to bet you have a HEAP (table without a clustered index). If so reindexing will not fix fragmentation for the Heap. Thanks for the sp_configure but to be really useful you need to show the advanced options.

--

Andrew J. Kelly SQL MVP

Re: Poor performance after upgrading to sql server 2005

"Rich Wood" <RichWood@xxxxxxxxxxxxxxxxxx> wrote in message

news:3BEFA6B5-F057-4BA3-BA9A-B8AEF97263F8@xxxxxxxxxxxxxxxxxx

The execution plans are not exactly the same -- drilling down into the execution plans shows that for the same step, the SQL 2005 db has a higher CPU cost. I suppose the units of measure may not be exactly the same. I can't change the MAXDOP since it's SQL 2005 standard. Disk IO is not the issue since the database fits in memory and looking at perfmon output, it's not even accessing the disk.

Digging into this a little deeper, I've found that minor differences in the t-sql result in huge differences in performance between the two databases. Most of these seem to be related to sub-queries. I don't mind replacing all the sub-queries in all of our database code but it would be a pain and I'd really like to understand exactly what is happening. By the way, when I look at the dm_db_index_physical_stats table I can see some fragmented indexes, but when I rebuild them with the ALTER INDEX statement and the REBUILD option there is no change in the index fragmentation reported in the physical table.

By the way, I have also run tests where the SQL 2005 db significantly outperforms the SQL 2000 db, for example, when replacing a sub-query with

Re: Poor performance after upgrading to sql server 2005

a
join or with a cached result in a table
variable.

Here is the sp_configure output:

```
allow updates 0 1 0 0
clr enabled 0 1 0 0
cross db ownership chaining 0 1 0 0
default language 0 9999 0 0
max text repl size (B) 0 2147483647 65536
65536
nested triggers 0 1 1 1
remote access 0 1 1 1
remote admin connections 0 1 0 0
remote login timeout (s) 0 2147483647 20
20
remote proc trans 0 1 0 0
remote query timeout (s) 0 2147483647 600
600
server trigger recursion 0 1 1 1
show advanced options 0 1 0 0
user options 0 32767 0 0
```

Thanks for your response.

Rich Wood

"Andrew J. Kelly" wrote:

Are you saying the
execution plan is identical
on both machines? Are
you
using the graphical
showplan to determine this
or some other way? Is
parallelism involved? Have
you played with MAXDOP?
What is your disk
configuration like and where
are the files? Can you show
the output
of
sp_configure?

--

Andrew J. Kelly SQL MVP

Re: Poor performance after upgrading to sql server 2005

"Rich Wood"

<RichWood@xxxxxxxxxxxxxxxxxxx>

wrote in message

news:3C36BDDF-4696-4B81-9743-4A5BB63C4F53@xxxxxxxxxxxxxxxxxxx

I have two machines, one running windows server 2000 and sql server 2000 standard and the other running windows server 2003 64-bit and sql server 2005 64-bit. The sql 2000 machine has two 2.4 GHz processors with 512 K cache and 4 GB memory and the sql 2005 machine has two dual core 2.8 GHz processors with 2 MB L2 cache and 12 GB memory.

I set up the same database on both machines

Re: Poor performance after upgrading to sql server 2005

and ran a series of production jobs to compare performance and in every case the sql 2000 db outperformed the sql 2005 db. Sometimes the difference was more than 20 times.

We also ran a series of benchmark tests to compare the cpu, memory and disk performance independent of sql server and in every case the sql 2005 machine significantly outperformed the sql 2000 machine.

Comparing actual execution plans between the two servers, it looks like the sql 2005 db uses more

Re: Poor performance after upgrading to sql server 2005

cpu than the
sql 2000 db
to perform
the same
operations.
We have
optimized
the indexes
and updated
the statistics
on
the
sql 2005
machine.
What else
could be
causing this
problem?

Thanks,
Rich Wood