

# Re: Performance question

---

*Source:*

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.fulltext/2005-05/msg00065.html>

---

- *From:* "Hilary Cotter" <[hilary.cotter@xxxxxxxx](mailto:hilary.cotter@xxxxxxxx)>
  - *Date:* Sun, 15 May 2005 21:16:44 -0400
- 

The approach you take for problems like this is to partition your tables, perhaps in your case by sales rank. For instance you might want to break your tables into 10 sub tables. One from 1-10, another from 11-20, etc.

Then limit each results set to 100 and union the results. This might end up more expensive than what you are currently experiencing.

--

Hilary Cotter

Looking for a SQL Server replication book?

<http://www.nwsu.com/0974973602.html>

Looking for a FAQ on Indexing Services/SQL FTS

<http://www.indexserverfaq.com>

"John" <[john36356@xxxxxxxxxxxxxxxxxxxx](mailto:john36356@xxxxxxxxxxxxxxxxxxxx)> wrote in message [news:ebYiMcaWFHA.2572@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:ebYiMcaWFHA.2572@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

> Hello,

> Our main product catalog is approx. 3.1 million rows, with a full-text index

> on 3 (varchar) columns. For the past year as our catalog has grown, we have

> experienced continuing performance degradation, to the point that we are

> looking at biting the bullet and migrating this application to Oracle Text,

> which from our initial testing is several orders of magnitude faster (we're

> going to stick with SQL Server for everything else). Obviously we'd like to

> avoid that due to cost issues. However, this application is very important

> and if we've reached the limit of SQL Server then so be it. Our problem is

> that when a customer searches our catalog, we sort the search results based

> on their sales rank which does not allow us to use the "top\_n" parameter of

> containstable or freetexttable. For example, say a customer searches our

## Re: Performance question

> catalog for a relatively common word that results in around 72,000 results  
> (takes approx 11 sec on subsequent runs...over 1 min on first run, which  
is  
> the most important statistic). If we were to use (say) n=2000 for the  
top\_n  
> parameter, our best selling products would not be returned from the FTS  
> engine. We need to return all the results so we can sort them by sales  
rank  
> and display them to customers.  
>  
> The server is running Windows 2003, with 6GB RAM, 16 x 15,000K RPM SCSI  
> drive in a RAID 10, in a dual opteron configuration the with transaction  
log  
> on a seperate RAID volume. SQL is SQL Server 2000, SP4 (I've included the  
> output of @@version below). Perfmon shows that the server isn't sweating  
at  
> all during these queries from a disk, memory, or CPU standpoint, so that  
> leaves SQL as the performance bottleneck. Our most recent population was  
> around a month ago, so the catalog is relatively up to date.  
>  
> We've fooled around with increasing the memory available to FTS, but that  
> did not seem to make a difference. Perhaps we did not do it right since  
> mssearch.exe is still only showing about 49,000K in memory—but since we  
> are using AWE this could be distorted.  
>  
> We're going to make one last gasp at improving the performance here before  
> dumping SQL Server and moving to Oracle. Help!  
>  
> John  
>  
> @@Version:  
> Microsoft SQL Server 2000 – 8.00.760 (Intel X86) Dec 17 2002 14:22:05  
> Copyright (c) 1988–2003 Microsoft Corporation Enterprise Edition on  
Windows  
> NT 5.2 (Build 3790: )  
>  
> Sample query:  
> SELECT distinct <field list>  
> FROM containstable(<ft-table>, <ft-field>,"<common term>") as ct  
> JOIN <ft-table> t with (nolock) on t.<PK>=ct.[key] //tables are 100%  
> readonly except during monthly updates, hence the nolocks  
> join salesRank sr with (nolock) on sr.<PK>=ct.[key]  
> order by sr.SalesRank  
>  
>

Re: Performance question

- **References:**

- ◆ **Performance question**

- ◇ From: John

- Prev by Date: **Performance question**
- Next by Date: **sqlfthndlr error/event id**
- Previous by thread: **Performance question**
- Next by thread: **sqlfthndlr error/event id**
- Index(es):
  - ◆ **Date**
  - ◆ **Thread**