

microsoft.public.sqlserver.fulltext: Re: Performance problems -- need guidance on scaling

Re: Performance problems -- need guidance on scaling

Source:

<http://www.tech-archive.net/Archive/SQL-Server/microsoft.public.sqlserver.fulltext/2004-11/0047.html>

From: Hilary Cotter (*hilary.cotter_at_gmail.com*)

Date: 11/06/04

Date: Fri, 5 Nov 2004 21:12:27 -0500

Hi Kevin

No, by caching queries I mean caching the search results so when a search on the same search phrase is repeated, the results are returned from the cache of search results as opposed from the catalog.

This requires you to pregenerate search results pages for each frequently issued or expensive search phrase. You could go so far as to pregenerate all previous searches into static pages of search results and refresh them on a daily or hourly basis.

When you query a list would be checked to see if the search results are pregenerate and if they are the client would be redirected to the static pages. If not, the search would be directed to the catalog.

Please refer to this link for an isapi extension that does this.

<http://groups.google.com/groups?hl=en&lr=&safe=off&selm=epmRvYYdDHA.2176%40TK2MSFTNGP09.phx.gbl>

and

<http://groups.google.com/groups?hl=en&lr=&safe=off&selm=eXGrNZYdDHA.2416%40TK2MSFTNGP09.phx.gbl>

Interestingly enough MSSearch (the engine which provides the search service for SQL FTS), relies on the file system cache to provide its caching. No queries per se are cached, but rather the catalog pages are "cached". One of the developers who worked on MSSearch told me that this works optimially for them, as opposed to any other cache scavanging mechanism. One of the reasons for this is the size of the database pages MSSearch uses.

--

Hilary Cotter

Looking for a SQL Server replication book?

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

"Kevin Kline [MVP]" <kevin.kline@quest.com> wrote in message
news:eHQulZ3wEHA.1400@TK2MSFTNGP11.phx.gbl...

> Hi Hilary,

>

Re: Performance problems -- need guidance on scaling

microsoft.public.sqlserver.fulltext: Re: Performance problems -- need guidance on scaling

> Is there an easy way, like a perfmon counter, to tell how many searches
hit
> a cache and how many don't?
>
> Thanks,
>
> -Kevin Kline
> Quest Software
> SQL Server MVP
>
> I support the Professional Association for SQL Server (PASS),
> www.sqlpass.org
>
> "Hilary Cotter" <hilary.cotter@gmail.com> wrote in message
> news:uAp1%236qwEHA.2908@tk2msftngp13.phx.gbl...
> > First off, the best thing you can do is to cache your queries, so that
> they
> > don't hit the search catalog at all. The problem with this approach is
> that
> > you will not have real time results, but there is nothing to stop you
from
> > incrementally generating the per generated html search results pages in
> > response to new content which is added. On one system where we did this
we
> > achieved a staggering 1400%
> >
> > So my question to you is can you cache queries, ie are many of your
> searches
> > on the same topics? If so such a caching scheme is excellent for this.
> >
> > Another option is to partition your tables into different catalogs so
that
> a
> > portion of your data exists in each catalog. MSSearch allocates threads
to
> > each catalog so there are performance advantages to implementing such
> > partitioning. If you can't break your data logically into different
groups
> > so you know a query will be satisfied with a single catalog you will
have
> to
> > do a union based query.
> >
> > Unfortunately SQL FTS benefits from multiple processors, with a sweet
spot
> > for 8 way processors. Using SQL FTS on a single proc box is not optimal
> even
> > with hyperthreading.
> >
> > Also while RAID 0 can offer better performance than RAID 1 or RAID 5 it
is
> > not fault tolerant, and in the long run may not be the best choice.
> >
> > Did you follow the recommendations in this article:
> >
> >
>
> <http://www.microsoft.com/downloads/details.aspx?FamilyID=824400af-315a-4c97-8eb2-a6047c42b222&dis>
> >
> > or this one?
> >
> >

microsoft.public.sqlserver.fulltext: Re: Performance problems -- need guidance on scaling

```
>
<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnsqpro04/html/sp04f9.asp>
> >
> >
> >
> >
> >
> > "Phil Sherrod" <phil.sherrod@REMOVETHISSandh.com> wrote in message
> > news:N76dnQOZqsxcFBfcrVn-gQ@giganews.com...
> > > We provide an indexing and searching service for our clients, and the
> > > searches are bogging down as the demand has increased. We need to
> > > increase
> > > the capacity of our system, so I am looking for some guidance.
> > >
> > > Current hardware:
> > >
> > > Single AMD Athlon 2800+ CPU
> > > 2.5 GB RAM
> > > 2 IDE SATA 10,000 RPM disks on Adaptec RAID controller in RAID 0
mode.
> > > 2 IDE IDE 7,200 RPM disks on Promise RAID controller in RAID 0 mode.
> > > 1 IDE 7,200 RPM system disk
> > >
> > > Both the SQL database and the full-text catalogs are on the Adaptec
RAID
> > > system because it is much faster than the old Promise system. We are
> using
> > > the Promise RAID system for our application data files.
> > >
> > > Current software:
> > >
> > > Windows 2000 Advanced Server
> > > SQL server 2000 version 8.00.760
> > > Language = us_english
> > >
> > > Database and index:
> > >
> > > Single table with about 15 M rows.
> > > Approximately 300,000 rows are added and removed every day. The
> > > additions
> > > and deletions are running non-stop day and night while the searches
are
> > > being done. We are using change tracking to manage the insertions
and
> > > deletions.
> > > We are indexing a single ASCII text column with an average of about
100
> > > characters and 7 words per row.
> > >
> > > Application:
> > >
> > > Our application receives search requests via a TCP/IP connection,
queues
> > > them internally and feeds them into the SQL searching system. Our
> > > application is multi-threaded, and we have it set so that it never
sends
> > > more than 4 simultaneous searches to the SQL server. During the busy
> > > times
> > > we sometimes have 30 to 100 pending searches waiting in line in the
> queue
> > > in
```

microsoft.public.sqlserver.fulltext: Re: Performance problems -- need guidance on scaling

> > > our application.
> > >
> > > We are using the CONTAINSTABLE method for full-text searching with a
> limit
> > > of 750 rows to be returned by CONTAINSTABLE. Virtually all of our
> > > searches
> > > have a series of AND clauses. Some of them are even more complex with
> > > parentheses and OR clauses.
> > >
> > > Some simple searches take only a couple of seconds, but we are seeing
> some
> > > of the more complex searches taking 20 to 30 seconds during light
> periods.
> > > During heavy periods when we have 4 simultaneous searches going, the
> > > search
> > > times are dragging out to 45 to 60 seconds which is unacceptable.
> > >
> > > When we were using the old, slow Promise RAID array for the database
and
> > > catalog, we were clearly I/O bound.
> > > With the newer and faster Adaptec RAID array, we see our CPU time
> pegging
> > > at
> > > 100% for extended periods.
> > >
> > > Search load:
> > >
> > > We are averaging one search every 5 seconds over a non-stop 24-hour
> period
> > > (including very slow times in the early morning). The demand (which
we
> > > cannot currently handle) is probably several searches per second
during
> > > the
> > > busy periods.
> > >
> > > In addition to the searches, approximately 800 simple, single-key (key
> > > type=long integer) queries are done to the SQL database per minute.
> > >
> > > Future needs and scalability:
> > >
> > > We see the search demands of our customers growing briskly over time.
> > > Within a year we may have 3 or more times the number of searches that
we
> > > currently have. So scalability is an important issue.
> > >
> > > What suggestions do you have for upgrades to our system to handle the
> > > current and anticipated future search needs?
> > >
> > > --
> > > Phil Sherrod
> > > (phil.sherrod 'at' sandh.com)
> > > <http://www.dtreq.com> (decision tree modeling)
> > > <http://www.nlreq.com> (nonlinear regression)
> > > <http://www.NewsRover.com> (Usenet newsreader)
> > > <http://www.LogRover.com> (Web statistics analysis)
> > >
> > >
> > >
> > >
> > >