

Re: Optimizing for speed

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.compactframework/2004-10/0156.htm>

From: Carlos Fernandez (*CarlosFernandez_at_discussions.microsoft.com*)

Date: 10/04/04

Date: Sun, 3 Oct 2004 22:23:08 -0700

Ryan,

Thanks for your quick reply. I'm going to read through your links carefully, but in the mean time, this is the typical situation in the application: Simple query (such as SELECT ID, CODE, STOCK FROM PRODUCTS) which returns 1100 records or so. The grid (which has room for like 10-11 rows) displays ID and CODE, while I use STOCK to display a msgbox if the user clicks on a sold-out product.

The query itself is fast, it's actually the loop that reads all the records and takes the time. I'd like to immediately display records and allow user interaction. My idea is to fetch the minimum number of records, display them, and load the rest as the user moves through the grid with the scrollbar but maybe there's another way.

Thanks.

"W.G. Ryan eMVP" wrote:

- > *Carlos:*
- >
- > *There's a lot of 'ifs' here that could be affecting things. The first thing*
- > *I'd consider is how much data am I loading in the grid vs. what the user*
- > *needs. If you're pulling out 1000 records but the user really only uses 10*
- > *most of the time, then query size would definitely be something to look at.*
- > *Another thing is how many trips to the db are you making? You have chioce I*
- > *of basically grabbing all the data and then storing the dataset as static*
- > *property for instance which will front load processing time but save time*
- > *over the long haul. In general, the as you need it approach is my leaning.*
- >
- > *As far as grids, you can't bind them to datareaders but if I just needed a*
- > *value or two, I'd definitely look to using one where possible. Indexes are*
- > *another thing. If you're really trying to squeeze out everythign that you*
- > *can get, set the BeginLoadData method of any given datatable before you*
- > *start loading the data*
- >

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfssystemdatadatatableclasstopic.asp>

> and then call EndLoadData

>

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemDataDataTableClassTopic.asp>

> ..I'd also shut of the EnforceConstraints property of the dataset if you have
> them

>

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpref/html/frlrfSystemDataDataSetClassTopic.asp>

-

> but don't leave them off of course, just set it back when you're done with
> the load. Lots less events will get fired if you do this – but I can't
> promise a huge performance increase because I don't know where the
> bottleneck is on your app. In some instances the differences are stunning,
> in others where this isn't the bottleneck the differences aren't worth
> mentioning.

>

> Those are the first things that come to mind, but if you could elaborate on
> what you mean by " put
> them in rows and display the grid " that may provide some insight into
> where the bottleneck is. At least the nice part of CE is that you don't
> have to worry about network congestion being the bottleneck ;-)
> "Carlos Fernandez" <CarlosFernandez@discussions.microsoft.com> wrote in
> message news:115783D8-1360-4CB3-91CE-A96A65EB9E05@microsoft.com...
> > Due to time constraint, my last project (first one in VB.NET for us, I
> might
> > add) was done 'quickly' rather than right, meaning that it is 'good
> enough'
> > for the client but not the kind of project one feels proud of after
> delivery.

> >

> > I happen to have some spare time on my hands which I plan to spend
> > optimizing the whole thing just to improve myself.

> >

> > One of the things I know we didn't do right, simple as it is, is the
> display
> > of SQL.CE based data on grids. We just execute a query, grab all records,
> put

> > them in rows and display the grid – slow as hell as you can easily guess.

> >

> > Given you have the data in one or several tables, what is the fastest way
> to

> > display it on a grid? Fastest as in 'fastest availability to the user', so
> > retrieving records as needed by user interaction is fine. This is actually
> my

> > first idea.

> >

> > Also, suppose you need one of the fields from the query not for display
> but

> > for something else (i.e. you can't just feed the results to the datagrid,
> > even if it's possible to do so, which I'm not sure).

> >

> > All suggestions welcome, I have the time to try them all.

microsoft.public.dotnet.framework.compactframework: Re: Optimizing for speed

> >
>
>
>