

Re: Button click with Remoting

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.remoting/2004-06/0137.html>

From: Sahil Malik (contactmethrumblog_at_nospam.com)

Date: 06/09/04

Date: Wed, 9 Jun 2004 09:46:17 -0400

Ok so lets say that one button is .. say your Save Button.

What I have in my project is, I call <<myremoteobject>>.Save ... and donot have to explicitly Disable the form or anything like that since the call is "Synchronous". That actually disables (sort of) my form just as if there were no remoting involved.

If you are performing an ASynch call instead, then I hope that first you have good reason for it and secondly you are not using Events from Server to Client for that communication .. instead you have the client check for the status of that call every now and then (more scaleable architecture). .. another solution is to have the client act as the server, and the server should then keep updating a property of an object to notify the client of the update. The big difference here is better lifetime control and connections going from server to client ... anyway before I digress from your point, here is back to your real question.

Now whether or not you disable the form in such a call, depends on your situation. In situations where I've had to make Asynch calls, I've done both.

For example, on the login screen, I disable the Login button until I get a response.

However, there is another situation where I have a progress bar running in a status bar displaying the status of a previously submitted request. In that I intend the user to continue doing other stuff, so I don't disable anything. There's a seperate thread running on the server doing that job.

– Sahil Malik

Independent Consultant

You can reach me thru my blog at –

<http://www.dotnetjunkies.com/weblog/sahilmalik/>

"John" <anonymous@discussions.microsoft.com> wrote in message news:1a55101c44e1f\$d424f100\$a001280a@phx.gbl...

>

> Sahil

>
> *OK, how about one button that performs an asynchronous*
> *remoting call, and then several other buttons that*
> *perform client side actions. Form also has tabs.*
>
> *Should I disable the entire form while the call is being*
> *made?*
>
> *Thanks,*
> *John*
>
>
> >-----Original Message-----
> >*John,*
> >
> >*If every button click is throwing a remoting call, you*
> >*are probably not*
> >*architecting the application right.*
> >
> >*Remoting emphasizes that you must use Chunky, not chatty*
> >*interfaces, viz.*
> >*think of the disconnected model ADO.NET and .NET*
> >*encourages, download all*
> >*that you might ever need as a dataset, or your custom*
> >*object, and do all the*
> >*button pressing, tab switching UI changes etc. on the*
> >*client, with the final*
> >*remoting call going when the user hits "Save".*
> >
> >*Therefore, instead of remoting every single call, you*
> >*now remote only two*
> >*calls.*
> >
> >*Why is this a good architecture, 6 years back, it took*
> >*1/2 a second to ping*
> >*tokyo, and it took 1 hour to download 1 MEG from Tokyo.*
> >*Today, it takes 1/2 a second to ping tokyo, but 2*
> >*seconds to download 1 MEG.*
> >
> >*Downloading tonnes extra is not a big deal, it is the*
> >*repeated network calls*
> >*that is a big deal.*
> >
> >*Hope that helps :)*
> >
> >– *Sahil Malik*
> >*Independent Consultant*
> >*You can reach me thru my blog at –*
> >*<http://www.dotnetjunkies.com/weblog/sahilmalik/>*
> >
> >

> >
> > "John" <anonymous@discussions.microsoft.com> wrote in
> message
> >news:17aa101c44969\$99d84de0\$a501280a@phx.gbl...
> >>
> >> *How should I handle the user clicking a button that
> calls
> >> an asynchronous SAO? Should I disable the button until
> >> my call returns? Just curious about what over people
> do.
> >>
> >> What about forms with lots of buttons and lots of tabs,
> >> should I disable everything until my call returns?
> >>
> >> Thanks in advance,
> >> John
> >
> >
> >.
> >*