

Cancelling an ADO Command takes too long

Source:

<http://coding.derkeiler.com/Archive/Delphi/borland.public.delphi.database.ado/2006-11/msg00145.html>

- *From:* "Clayton Arends" <nospam_claytonarends@xxxxxxxxxxx>
 - *Date:* Wed, 22 Nov 2006 09:39:50 -0800
-

I have created an application to test asynchronous operations in ADO to improve performance of one of my products. Mostly I'm doing this so that the user is able to cancel long-running or runaway queries without having to use task manager.

The problem I have is that the cancel operation doesn't appear to be cancelling the query at all. The amount of time that the query I am using takes is around 14 seconds. This same amount of time passes even if I attempt to cancel the query. As tests I have used a longer running query and reduced the CommandTimeout of both the command and the connection without a change in results. I also tried all combinations of the eoAsyncXXX options in TADOCommand.ExecuteOptions (aside from the starting point which I mention below).

Configuration

I added a TADOConnection to the form and set ConnectOptions to [coAsyncConnect]. I am using an Oracle 10G connection at my office (using an Oracle OLEDB driver on my computer). I added a TADOCommand to the form and set ExecuteOptions to [eoAsyncExecute]. I added a TEdit to specify a SQL statement and a TMemo which receives status information. I added a TButton which executes the command or, if a command is executing, cancels a command:

```
{ My program is written in BCB6 but I've converted to Delphi for this
post. Any typos are most likely from the conversion to Delphi code
without compiling (IOW, please look past basic typos and look at the
meat) }
```

```
if (stExecuting in ADOCommand1.States) then
begin
Memo1.Lines.Add('Cancelling ...');
Memo1.Refresh();
ADOCommand1.Cancel();
Memo1.Lines.Add('Cancelled');
exit;
end;
```

Cancelling an ADO Command takes too long

```
ADOCCommand1.CommandText := Edit1.Text;  
ADOCCommand1.Execute();
```

```
Memo1.Lines.Add("Executing ...");
```

So here is what I need to know. Have I configured my test correctly to use asynchronous ADO? Is Cancel() always a blocking operation? Is Cancel() optionally supported by the database? If the query is a runaway query should Cancel() eventually come back?

I have performed some general archive searches on this subject. I found a few cases where the authors mentioned that Cancel() has a long delay but none of them mentioned that it was the same amount of time as the normal running query. Most other posts I found were simply questions regarding how to cancel long running queries. The answer in those cases was to use asynchronous ADO.

Thank you for any insight,

– Clayton

PS – This post was originally in borland.public.cppbuilder.database.ado but it hasn't generated any interest. My hope is that this more active ADO newsgroup can shed some light on this topic.

.