

# Re: Working Transactions somehow started not to work

---

*Source:*

<http://www.tech-archive.net/Archive/Access/microsoft.public.access.adp.sqlserver/2006-10/msg00100.html>

---

- *From:* "aaron.kempf@xxxxxxxx" <aaron.kempf@xxxxxxxx>
  - *Date:* 16 Oct 2006 10:49:35 -0700
- 

I find this hilarious

If you are using JET, than you are not expected to consider that performance might be important.

I mean seriously.. If Microsoft really feels that way then they need to fuck themselves.

HIGH-PERFORMANCE, SIMPLE DATA ENTRY AND REPORTING.  
That is what we DEMAND.

-Aaron

Sylvain Lafontaine (fill the blanks, no spam please) wrote:

Sorry, but after re-reading myself, I see that what I wrote is not clear at all; it should have been:

Transaction and locking are not properties of recordsets but are properties of connections and as such, they are not necessarily associated with server-based cursors or with client-side or server-side recordsets. Opening a served-based cursor will put a lock on the table but this is not the only way of having a lock; as in fact you can even have a lock without any cursor at all.

If you want to, you can have a client based cursor that will lock the records on the server. The only restriction would be that the recordset must remains connected because locks on SQL-Server are always associated with an open connection. If a connection is closed, any opened lock are released and any uncommitted transaction are rolled back automatically.

Re: Working Transactions somehow started not to work

If Access doesn't use lock with its recordsets, it's not because of a technical impossibility but simply because it's not programmed to act in this way. (Doing so would be a major performance sink but that's another story). Don't confuse the capabilities of Access/ADP with the capabilities of ADO or of SQL-Server.

When using a JET database backend, you can easily have a recordset with pessimistic locking in Access but this is not because of a technical superiority of JET; it's simply because MS doesn't feel that any performance penalty can be considered as important when you are using JET as the backend. If you are using JET, than you are not expected to consider that performance might be important.

--

Sylvain Lafontaine, ing.  
MVP – Technologies Virtual-PC  
E-mail: <http://cerbermail.com/?OugbLEWINF>

"Vadim Rapp" <vr@xxxxxxxxxxxxxxxxxxxxxx> wrote in message  
[news:u3%235jr7GHA.3452@xxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:u3%235jr7GHA.3452@xxxxxxxxxxxxxxxxxxxxxxxxxx)

Hello Sylvain,  
You write in conference microsoft.public.access.adp.sqlserver on Wed, 11  
Oct 2006 10:47:43 -0400:

SL> Transaction and locking are not necessary are properties of  
connections

Locking is a property of the recordset.

SL> and are not associated necessarily with server-based cursors.

I think they are. Client-based cursor, by definition, does the following:

1. download the records
2. edit the records "offline"
3. upload the records back to the database, with checking first if they have not changed since (1)

During (2), the database does not "know" that the records are being edited, there are no locks that would prevent others from touching the downloaded records. The client may even disconnect from the database (disconnected recordset). The database lock is established only during (1) and (2)

Server-based cursor, contrary, does establish database lock once the record is edited.

SL> In fact, you can have an open lock on a row, page or table without  
SL> having any cursor at all!

Re: Working Transactions somehow started not to work

With recordsets? how? (I understand that it's possible by non-recordset-returning operations like UPDATE TABLE, but we are talking here about Access forms).

Vadim Rapp

SL> (This is why so many people have trouble with dead-locks sometimes).

SL> --

SL> Sylvain Lafontaine, ing.

SL> MVP – Technologies Virtual-PC

SL> E-mail: <http://cerbermail.com/?QugbLEWINF>

SL> "Vadim Rapp" <vr@xxxxxxxxxxxxxxxxxxxxxx> wrote in message

SL> [news:O\\$M\\$5yC7GHA.2364@xxxxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:O$M$5yC7GHA.2364@xxxxxxxxxxxxxxxxxxxxxxxxxxxx)

SL>> Hello Sylvain,

SL>>

SL>> I think, in a nutshell, Access can be assumed incompatible with  
SL>> transactions. For the simple reason: transactions and locking involve  
SL>> server-based cursors. While Access (as well as vs.net) works with  
SL>> client-based cursors. Client-based cursors create their own  
SL>> pseudo-transaction; it may be good or bad, but it's totally different  
SL>> from the server-based cursors, hence from locking, hence from  
SL>> transactions.

SL>>

SL>> I totally agree, however, with the thought that "true" locking is  
SL>> hardly compatible with interactive work. A user who is trying to edit  
SL>> a record currently locked by someone else, will certainly assume that  
SL>> his application or even computer has locked up. To be useful, the  
SL>> locking mechanism should include some statement that would attempt  
to

SL>> place the lock, and would be asynchronous, so the user would be able  
SL>> to cancel the wait for the resource.

SL>>

SL>> When I found that mentioned workaround with creating the true  
SL>> transaction by using connection.execute "begin transaction", I  
SL>> actually opened support ticket with Microsoft and asked if it's  
SL>> supported scenario (having pretty good idea what would be the  
answer),

SL>> and why begintrans does not work as expected. The response was that,  
SL>> indeed, at some point there were plans to make begintrans etc. fully  
SL>> working in ADP, but then it was stopped in the middle, so the whole  
SL>> thing remains half-baked and undocumented.

SL>>

SL>> In all new programming environments, Microsoft is actually pushing  
SL>> client-based cursors, and there are results indeed: it looks like now  
most

SL>> developers don't even consider true locking. Recently, I tested this  
SL>> hypothesis by asking moderately difficult locking-related question on

Re: Working Transactions somehow started not to work

SL>> experts-exchange – nobody even tried to answer along the correct lines  
SL>>  
([http://www.experts-exchange.com/Databases/Microsoft\\_SQL\\_Server/O\\_22012456.html](http://www.experts-exchange.com/Databases/Microsoft_SQL_Server/O_22012456.html)) – though there are \_very\_ skilled sql programmers there.  
I  
SL>> won't be too surprised if in some future version of sql server,  
SL>> user-controlled locking will be abandoned as such.  
SL>>  
SL>> Vadim Rapp  
SL>>  
SL>> You wrote in conference microsoft.public.access.adp.sqlserver on Tue,  
SL>> 10 Oct 2006 00:18:26 -0400:  
SL>>  
SL>>> Personally, I've always been against the use of transactions for  
SL>>> long term editing and every time I see someone implementing a cancel  
SL>>> function for a form by using a Rollback encased in a transaction make  
SL>>> me grind my teeth. The problems with scaling and performance  
SL>>> degradation that this can lead to are easy to guess.  
SL>>  
SL>>> However, even if I'm against their use in this context, this doesn't  
SL>>> explain why his particular piece of code is not working properly.  
SL>>> Instead of using one of the connection created by Access, he creates  
SL>>> and use its own connection (the line « ConnectionString =  
SL>>> CurrentProject.Connection » doesn't begin wit the SET command, so  
SL>>> instead of reusing the CurrentProject.Connection object, he's  
SL>>> creating a new object using the connection string of  
SL>>> the CurrentProject.Connection object) so theoritically, this should  
SL>>> work.  
SL>>  
SL>>> However, by using this connection string, he's also using the  
special  
SL>>> OLEDB driver created for ADP and that is standing between the  
bound  
SL>>> forms and SQL-Server. Even with a separate connection string, the  
SL>>> effect of using an open transaction on this driver is totally  
SL>>> undocumented by Microsoft and in my opinion, totally unpredictable.  
SL>>> It's clear in my opinion that even with a separate connection, this  
SL>>> driver will try to open and close its own transactions while  
updating  
SL>>> the database for saving the edited data on the bound forms.  
SL>>  
SL>>> Maybe he could try following the @@transaction count on the server  
SL>>> (inside a stored procedure) to solve this problem but in my opinion,  
SL>>> opening a direct connection to the SQL-Server without using the  
SL>>> connection string of the CurrentProject.Connection object (to make  
SL>>> sure  
SL>>> that only the OLEDB provider for SQL-Server is used, not that

Re: Working Transactions somehow started not to work

strange

SL>>> beast called the Microsoft OLEDB provider for Access) and using

SL>> unbound

SL>>> forms should be a much more safer approach (albeit also more

SL>>> complicated but that's life) than the previously posted code.

SL>>

SL>>> In the past, Microsoft has warned many times that trying to use

SL>>> transactions with bound forms would have a high probability of

SL>> problems

SL>>> and failures; I don't see any reason to doubt their parole. If you

SL>> use

SL>>> them and don't have any problem, good for you but if you are

unlucky,

SL>>> then you are on your own.

SL>>

SL>>> --

SL>>> Sylvain Lafontaine, ing.

SL>>> MVP – Technologies Virtual-PC

SL>>> E-mail: <http://cerbermail.com/?QugbLEWINF>

SL>>

SL>>> "Vadim Rapp" <vr@xxxxxxxxxxxxxxxxxxxxxx> wrote in message

SL>>> [news:%23\\$7ajhA7GHA.1560@xxxxxxxxxxxxxxxxxxxxxxxxxx](mailto:news:%23$7ajhA7GHA.1560@xxxxxxxxxxxxxxxxxxxxxxxxxx)

SL>>>> Hello Ömer,

SL>>>>

SL>>>> you probably will be surprised, or maybe even won't believe, but  
the

SL>>>> code you posted could not work ever; at least, reliably. Unlike  
MDB,

SL>>>> with ADP, begintrans, committrans, and rollback are practically

SL>>>> ignored. See for example <http://support.microsoft.com/kb/223213> –

SL>> it's easy to see that the results would

SL>>>> be unpredictable.

SL>>>>

SL>>>> Also see

SL>>>>

SL>>

[http://www.eggheadcafe.com/aspnet\\_answers/AccessadpSQLServer/Jan2006/post25750796.asp](http://www.eggheadcafe.com/aspnet_answers/AccessadpSQLServer/Jan2006/post25750796.asp)

–

SL>>>> Sylvain's comment. There's workaround, but it's unsupported and  
also

SL>>>> may bring unpredictable results – see my comment in the end of the

SL>>>> above page.

SL>>>>

SL>>>> From all my experience of working with ADP, the best results are

SL>>>> achieved by allowing Access to manipulate the data its own way,

SL>> rather

SL>>>> than trying to force it to do what you want. For example, it's

SL>>>> better to specify recordsource for a form rather than

SL>>>> set me.recordset=<pre-created

recordset>>>> – i.e. allow Access to build the recordset as it wants,

SL>> rather

Re: Working Transactions somehow started not to work

SL>>>

SL>>>> than force it to use yours. The path of forcing your way on Access  
SL>>>> is paved with numerous hours of troubleshooting unpredictable  
SL>>>> behaviours exactly like what you just encountered.

SL>>>>

SL>>>> The best tool for working with transactions and proper locking is

SL>> VB6.

SL>>>> Everything else (Access, and VS.Net) is using client-based cursors.

SL>>>>

SL>>>> Vadim Rapp

SL>>>>

SL>>>> You wrote in conference microsoft.public.access.adp.sqlserver on

SL>> Fri,

SL>>>> 6 Oct 2006 13:57:18 +0300:

SL>>>>

ÖA>>>>> Dear friends,

SL>>>>

ÖA>>>>> I encountered such a problem and do not know how to correct it:

SL>>>>

ÖA>>>>> I am using XP Professional, Office 2003 (SP2) and SQL2000

SL>>>>

ÖA>>>>> I'd like to use a transaction on one to many form. Specifically I

ÖA>>>>> have Bank(main form) and Branches (Subform) These are  
designed as

ÖA>>>>> bound forms to TB\_BANK & TB\_BRANCH tables respectively.

However in

ÖA>>>>> order to use a transaction I open another connection and with this

ÖA>>>>> connection I start a transaction (Code as follows) at the open

ÖA>>>>> event of the main form. When user chooses to save whatever he  
has

ÖA>>>>> done

SL>> with

ÖA>>>>> the form I commit otherwise rollback.

SL>>>>

ÖA>>>>> Up until this morning this approach was working perfectly. While

ÖA>>>>> debugging on the run Access hanged couple of times. Even

though

ÖA>>>>> I recovered thru 'Compact & Repair Project' my ability to rollback

ÖA>>>>> has gone. Assuming that sth has happened to db I run SQL Query

ÖA>>>>> analyzer and tried updating tables under transaction and no

ÖA>>>>> problem. I have other adp's using the very same approach they  
also

ÖA>>>>> work fine.

SL>>>>

ÖA>>>>> To eliminate the problem I created a new adp and copied only the

ÖA>>>>> above forms & code behind it and still the same thing. For any  
one

ÖA>>>>> who could propose sth the code is as follows:

SL>>>>

ÖA>>>>> Private mIngOpMode As openMode

ÖA>>>>> Private mConnMain As New ADODB.Connection

Re: Working Transactions somehow started not to work

```
ÖA>>>>> Private mrstBank As New ADODB.Recordset
ÖA>>>>> Private mrstBranch As New ADODB.Recordset
SL>>>>
ÖA>>>>> Private Sub Form_Open(Cancel As Integer)
ÖA>>>>> Dim strSqlMain As String
ÖA>>>>> Dim strSQLSub As String
ÖA>>>>> Dim strFilter As String
ÖA>>>>> Dim lngCallerForm As Long
ÖA>>>>> Dim strWhere As String
SL>>>>
ÖA>>>>> Dim varOparg As Variant
ÖA>>>>> Dim strNewData As String
SL>>>>
ÖA>>>>> ' Format of strOpenArg:
ÖA>>>>> ' openMode;NewData;CallerField;CallerForm;Filter
SL>>>>
ÖA>>>>> Me.ServerFilter = ""
SL>>>>
ÖA>>>>> varOparg = Split(Me.OpenArgs, ";")
ÖA>>>>> Me.Caption = CStr(varOparg(0)) ' Caption
ÖA>>>>> mlngOpMode = CLng(varOparg(1)) ' Open Mode
ÖA>>>>> strNewData = CStr(varOparg(2)) ' New Data
ÖA>>>>> ' Caller Field
ÖA>>>>> ' Caller Form
ÖA>>>>> strFilter = CStr(varOparg(5)) ' Filter
SL>>>>
ÖA>>>>> With mcnnMain
ÖA>>>>> .ConnectionString = CurrentProject.Connection
ÖA>>>>> .Open
ÖA>>>>> .BeginTrans
ÖA>>>>> End With
SL>>>>
ÖA>>>>> ' MainForm Recordsource
ÖA>>>>> strSqlMain = "SELECT * FROM AYZ_TB_BANK WHERE " &
strFilter
ÖA>>>>> ' SubForm Recordsource
ÖA>>>>> strSQLSub = "SELECT * FROM AYZ_TB_BRANCH WHERE " &
& strFilter
SL>>>>
ÖA>>>>> mrstBank.Open strSqlMain, mcnnMain, adOpenKeyset,
ÖA>>>>> adLockOptimistic mrstBranch.Open strSQLSub, mcnnMain,
ÖA>>>>> adOpenKeyset,
SL>>>>> adLockOptimistic
SL>>>>
ÖA>>>>> Set Me.Recordset = mrstBank
ÖA>>>>> Set Form_AYZ_SUBFRM_BRANCH.Recordset = mrstBranch
SL>>>>
ÖA>>>>> Private Sub cmdCancel_Click()
ÖA>>>>> Dim strMsg As String, userResponse As VbMsgBoxResult
ÖA>>>>> Dim strWhere As String
ÖA>>>>> Dim strSQL As String
```

Re: Working Transactions somehow started not to work

```
SL>>>>
ÖA>>>>> On Error GoTo Err_cmdCancel_Click
SL>>>>
ÖA>>>>> If Me.cmdSave.Enabled Then
ÖA>>>>> strMsg = "Yaptıđýnýz deđipiklikler iptal edilecektir."
ÖA>>>>> userResponse = DisplayMessage(strMsg, vbExclamation +
ÖA>>>>> vbOKCancel)
SL>>>>
ÖA>>>>> If userResponse = vbOK Then
ÖA>>>>> mcnnMain.RollbackTrans
ÖA>>>>> Else
ÖA>>>>> GoTo Exit_cmdCancel_Click
ÖA>>>>> End If
ÖA>>>>> End If
ÖA>>>>> DoCmd.Close acForm, Me.Name
SL>>>>
ÖA>>>>> Exit_cmdCancel_Click:
ÖA>>>>> Exit Sub
ÖA>>>>> Err_cmdCancel_Click:
ÖA>>>>> DisplayMessage Err.Number & ": " & Err.Description, vbCritical
ÖA>>>>> + vbOKOnly, "HATA!" Resume Exit_cmdCancel_Click
SL>>>>
ÖA>>>>> End Sub
SL>>>>
ÖA>>>>> Private Sub cmdSave_Click()
ÖA>>>>> Dim strMsg As String, lngBankID As Long, strWhere As String
ÖA>>>>> On Error GoTo Err_cmdSave_Click
SL>>>>
ÖA>>>>> If IsNull(Me.txtBankCode) Then
ÖA>>>>> strMsg = "Banka Kodu' bođ olamaz."
ÖA>>>>> DisplayMessage strMsg, vbOKOnly + vbCritical, "UYARI!"
ÖA>>>>> Me.txtBankCode.SetFocus
ÖA>>>>> GoTo Exit_cmdSave_Click
ÖA>>>>> End If
SL>>>>
ÖA>>>>> mcnnMain.CommitTrans
SL>>>>
ÖA>>>>> DoCmd.Close acForm, Me.Name
ÖA>>>>> Exit_cmdSave_Click:
SL>>>>
ÖA>>>>> Exit Sub
ÖA>>>>> Err_cmdSave_Click:
ÖA>>>>> MsgBox Err.Description
ÖA>>>>> Resume Exit_cmdSave_Click
ÖA>>>>> End Sub
SL>>>>
ÖA>>>>> Private Sub Form_Close()
ÖA>>>>> Form_AYZ_FRM_BANKBR.txtFindCriteria = Me.txtBankID
ÖA>>>>> ' A little bit of house keeping
ÖA>>>>> Set mrstBank = Nothing
ÖA>>>>> Set mrstBranch = Nothing
```

Re: Working Transactions somehow started not to work

Re: Working Transactions somehow started not to work

```
ÖA>>>>> Set mcnnMain = Nothing
ÖA>>>>> End Sub
SL>>>>
ÖA>>>>> Private Sub Form_Dirty(Cancel As Integer)
ÖA>>>>> cmdSave.Enabled = True
ÖA>>>>> End Sub
SL>>>>
SL>>>> Vadim
SL>>>> -----
SL>>>> Vadim Rapp Consulting
SL>>>> SQL, Access, VB Solutions
SL>>>> 847-685-9073
SL>>>> www.vadimrapp.com
SL>>>
SL>>> Vadim
SL>>> -----
SL>>> Vadim Rapp Consulting
SL>>> SQL, Access, VB Solutions
SL>>> 847-685-9073
SL>>> www.vadimrapp.com
```

Vadim

-----  
Vadim Rapp Consulting  
SQL, Access, VB Solutions  
847-685-9073  
www.vadimrapp.com