

Re: Stored Procedures & Parameters Insert Problem

Source: <http://www.tech-archive.net/Archive/Data/microsoft.public.data.ado/2007-03/msg00055.html>

- *From:* incubusaurus@xxxxxxxxxx
 - *Date:* 14 Mar 2007 01:23:28 -0700
-

Hi Bob,

Thanks for your quick reply. I deleted my original message shortly after posting it, and posted a modified version, with some more detail, in microsoft.public.vb.database.ado. I have therefore included the modified post at the end of this post.

I have responded to your questions below.

On 13 Mar, 11:09, "Bob Barrows [MVP]" <reb01...@xxxxxxxxxxxxxxxxxx> wrote:

incubusau...@xxxxxxxxxx wrote:

Hi,

I am using ADO, in a fairly simple VB 6.0 client based database application, to manipulate an Access database. When I use a stored procedure with parametrised inputs to insert multiple records into a table, I end up with the same values repeated for each record, despite changing the value of the parameter. I suspect that somewhere along the line, the parameter value is being cached. To simplify this post, I have written a small procedure (see below) which produces the same results as the application.

<snip>

```
adoDBConn.Open "Provider=MSDataShape;" & _
```

Why use the MSDataShape provider? It is not needed for this activity. If your application has need for the functionality provided by this provider, You should consider using a separate Connection for that activity.

Re: Stored Procedures & Parameters Insert Problem

I use data shaping elsewhere in the application. I have created a class in the application to handle database functions, and the connection is opened as part of the class initialization. I included it here as I wanted the supplied code sample to be as true a representation of the application code as possible. I have tried the sample code both with and without the MSDataShape provider, and still get the same results.

```
"Data Provider=Microsoft.Jet.OLEDB.4.0;" & _  
"Data Source=" & App.Path & "\Test.mdb"
```

```
adoCommand.ActiveConnection = adoDBConn
```

```
On Error Resume Next  
adoDBConn.Execute "DROP PROCEDURE sp_InsertCount"
```

Why drop and recreate this procedure? It is contrary to the purpose of using stored procedures: create it once and re-use it.

This is so that I can easily change the stored procedure in the sample code, without having to manually delete it. This code is only in the sample provided here, and is not used in the application.

```
On Error GoTo 0
```

```
strCmd = "CREATE PROCEDURE sp_InsertCount AS "
```

This is now the time to get out of the habit of using "sp_" to prefix your stored procedure names. It is not hurting you in Jet, but it will definitely impair the efficiency of your procedures in SQL Server, which uses that prefix to identify system stored procedures. When told to execute a query with that prefix, SQL Server will waste time looking for the procedure in the Master database, only looking for it in the current database when it is not found in Master. Also, if you inadvertently give the procedure the same name as a system stored procedure, you may find yourself trying to figure out errors due to the attempted execution of the wrong procedure.

Okay, thanks for that. I will bear that in mind. Whilst trawling the

Re: Stored Procedures & Parameters Insert Problem

net for a solution, I remember reading this elsewhere, and have already made this change in my sample code.

```
strCmd = strCmd & "INSERT INTO TestCount (lngTestCount) "  
strCmd = strCmd & "VALUES (@lngTestCount)"
```

```
adoDBConn.Execute strCmd
```

```
adoCommand.CommandText = "sp_InsertCount"  
adoCommand.CommandType = adCmdStoredProc
```

```
Set prmADOParam = adoCommand.CreateParameter _  
("@lngTestCount", adInteger, adParamInput, , Null)
```

```
adoCommand.Parameters.Append prmADOParam
```

```
' ** This produces 20 rows all with the same value **  
For lngTestCount = 1 To 20
```

```
adoCommand.Parameters("@lngTestCount").Value = lngTestCount
```

It will be more efficient to use the prmADOParam object you went to such trouble to create instead of going through the Parameters collection here.
prmADOParam .Value=lngTestCount

In desperation, I have tried both methods in the sample code, but neither made any difference to the result. I take on board your comment about efficiency though.

I would experiment with

- 1) changing the Provider to the Jet provider, eliminating the MSDataShape provider. If that does not help:
- 2) setting the Command's Prepared property to True or False to see which setting solves the issue.

Re: Stored Procedures & Parameters Insert Problem

I have experimented with the Prepared property, and, as mentioned above, changing the Provider, both with no change to the result.

—
Microsoft MVP – ASP/ASP.NET

Please reply to the newsgroup. This email account is my spam trap so I don't check it very often. If you must reply off-line, then remove the "NO SPAM"

Since originally posting, I have also tried removing the parameter declaration from the code, and instead passing the values to the Execute command as an array. I still get the same results. If I had any hair to pull out, I would be pulling my hair out by now!

The missing part of the original post is as follows:

I am using ADO, in a fairly simple VB 6.0 client based database application, to manipulate an Access database. When I use a stored procedure with parametrized inputs to insert multiple records into a table, I end up with the same values repeated for each record, despite changing the value of the parameter. I suspect that somewhere along the line, the parameter value is being cached. To simplify this post, I have written a small procedure (see below) which produces the same results.

The test database contains one table, TestCount, in which there are two fields: an AutoNumber primary key field, and lngTestCount. Using the stored procedure, twenty records are created, all with lngTestCount set to 1. The AutoNumber field increments as expected, and the debug print of the parameter value shows the parameter value to be incrementing also.

When I run exactly the same SQL INSERT with the CommandType set to adCmdText instead of adCmdStoredProc, it works correctly.

I have tried the code:

- With and without the BeginTrans and CommitTrans commands, both inside and outside the loop.
- With different parameter names, with and without the @ symbol.
- With random numbers instead of incrementing numbers.
- With string data instead of numeric data.
- With and without the adExecuteNoRecords Execute option.
- With and without the MSDDataShape provider (which I use elsewhere in the application).
- With the Command's Prepared property set to True and False.

Re: Stored Procedures & Parameters Insert Problem

I have also spent many hours searching the web for an answer.

Can anyone help? I am sure I must be missing something very obvious.

```
Public Sub TestInsert()
Dim adoDBConn As ADODB.Connection
Dim adoCommand As ADODB.Command
Dim prmADOParam As ADODB.Parameter
Dim strConn As String
Dim strDBPath As String
Dim strCmd As String
Dim lngTestCount As Long

Set adoDBConn = New ADODB.Connection
Set adoCommand = New ADODB.Command

adoDBConn.Open "Provider=MSDataShape;" & _
"Data Provider=Microsoft.Jet.OLEDB.4.0;" & _
"Data Source=" & App.Path & "\Test.mdb"

adoCommand.ActiveConnection = adoDBConn
On Error Resume Next
adoDBConn.Execute "DROP PROCEDURE sp_InsertCount"
On Error GoTo 0

strCmd = "CREATE PROCEDURE sp_InsertCount AS "
strCmd = strCmd & "INSERT INTO TestCount (lngTestCount) "
strCmd = strCmd & "VALUES (@lngTestCount)"

adoDBConn.Execute strCmd

adoCommand.CommandText = "sp_InsertCount"
adoCommand.CommandType = adCmdStoredProc

Set prmADOParam = adoCommand.CreateParameter _
("@lngTestCount", adInteger, adParamInput, , Null)

adoCommand.Parameters.Append prmADOParam

' ** This produces 20 rows all with the same value **
For lngTestCount = 1 To 20
adoCommand.Parameters("@lngTestCount").Value = lngTestCount
Debug.Print adoCommand.Parameters("@lngTestCount").Value
'adoDBConn.BeginTrans
adoCommand.Execute
'adoCommand.Execute , , adCmdStoredProc Or adExecuteNoRecords
'adoDBConn.CommitTrans
Next

strCmd = "INSERT INTO TestCount (lngTestCount) "
```

Re: Stored Procedures & Parameters Insert Problem

```
strCmd = strCmd & "VALUES (@IngTestCount)"

adoCommand.CommandText = strCmd
adoCommand.CommandType = adCmdText

' ** This produces 20 rows all with incrementing values **
For lngTestCount = 1 To 20
adoCommand.Parameters("@IngTestCount").Value = lngTestCount
Debug.Print adoCommand.Parameters("@IngTestCount").Value
'adoDBConn.BeginTrans
adoCommand.Execute
'adoCommand.Execute , , adCmdStoredProc Or adExecuteNoRecords
'adoDBConn.CommitTrans
Next

adoDBConn.Close

Set prmADOParam = Nothing
Set adoCommand = Nothing
Set adoDBConn = Nothing
End Sub
```