

# Re: Huge memory consumption of ADODB Connection object

---

*Source:* <http://www.tech-archive.net/Archive/Data/microsoft.public.data.ado/2007-11/msg00078.html>

---

- *From:* "Bob Barrows [MVP]" <[reb01501@xxxxxxxxxxxxxxxxxx](mailto:reb01501@xxxxxxxxxxxxxxxxxx)>
  - *Date:* Tue, 27 Nov 2007 08:52:09 -0500
- 

Either I am missing something, or there is something you have left out, because what you have described sounds like something I handle every day by using a staging table for the imported data, an UPDATE statement (to update the ones that exist) and an INSERT statement to insert the ones that don't exist.

Again, unless the cursor is returning data to the client\* I don't see how this could have an impact on the client.

You say that the Connection object is growing ... how have you determined this? Why are you so sure it's the connection object rather than the recordset object that is growing?

\* which it very well may be in Sybase – in SQL Server, if one does not use SET NOCOUNT OFF in a procedure, informational messages ("x rows were effected") from each DML statement (UPDATE, INSERT, DELETE) are sent to the client as resultsets. I suppose this could cause some extra memory usage on the client, but I can't see it amounting to a lot, since these are server-side cursors that are only sent to the client when requested via NextRecordset. Anyways, Is there a similar behavior in Sybase? SQL Server was originally based on that rdbms, so there very well may be.

Kevin LZJ wrote:

Hi, Bob,

It's a equipment management system, in which there are some equipment in each room. So we have a DB table named as "ROOM", and a table named as "Equipment". For each equipment, we give it a name. When porting data from Excel file into SYBASE DB, we need check there is a equipment with same name existing in the same room. For example, name 'ABC1#', 'ABC01', 'ABC1', we think they are the same. Because this rule cannot be done in a single SQL statement, so we use a stored procedure to do this (check name the same), and we use a cursor to get each equipment's name in the same room, then call this SP to check they are the same. For my knowledge, I couldn't find a easy way (not using cursor to do this). The way we use SP in VB is rather simple :

## Re: Huge memory consumption of ADODB Connection object

```
Dim myConn as new ADODB.Connection
Dim myRecordset as ADODB.RecordSet
Dim strSQL As String
```

```
strSQL = "exec SP_PORT_DATA ..."
```

```
Set myRecordSet = myConn.Execute strSQL
Some processing
myRecordSet.Close
Set MyRecordSet = Nothing
```

In the code, the SP returns only one row to show the the result or data porting.

"Bob Barrows [MVP]" wrote:

I would be willing to bet that, with some thought, your complicated rules could be accomplished without a cursor. However, I'm in no position to try to win that bet.

As far as the server-based cursor affecting the Connection object, unless results are being returned to the client via the cursor, there should be no impact on the client from using a server-based cursor in a stored procedure. Additionally, a Connection object is not a "data container", it's more of a conduit between the database and an ADO data-receiver. So I'm unaware of any reason for your problem.

Perhaps you could show us how you call that stored procedure ...

Kevin LZJ wrote:

Hi, Bob,

Thanks for your kind reply.  
However, in my case, it's impossible to check whether same data exists only by using "if exists (select ...)", I need to apply some complicated rules for checking data, that's why I use cursor to get each row. My question is: the procedure runs in Sybase Server, why it makes ADODB Connection object consume so much memory (in my understanding, ADODB acts as a client ). Moreover, I first must determine this problem come from ADODB (Microsoft), or

Re: Huge memory consumption of ADODB Connection object

SYBASE.

"Bob Barrows [MVP]" wrote:

Kevin LZJ wrote:

controlHi all,

I encountered such a problem: ADODB Connection consumes huge memory before it is closed. I use VB to port some data from EXCEL file into SYBASE database. For some reason, I first convert the EXCEL file to DBASE format, then use ADO Data Control to get data from DBASE file, finally, I use ADODB to insert data into SYBASE database (using ADODB provider from SYBASE corp). The problem is, I use a database stored procedure to insert data in which there is a cursor to check whether same data is existing. If the content of cursor is more than 300 items, I found the memory used increases greatly (in my case, there are about 600 data items in the cursor, causing the memory used up to more than 2GB and getting error for the lack of memory). After some research, I found 2 points :

- 1). If I delete the cursor from the Stored Procedure, my app

Re: Huge memory consumption of ADODB Connection object

uses memory normally.  
2). Only when the ADODB connection is closed, can the huge memory used be freed.

So, my question is: why the ADODB connection object uses such huge memory even the stored procedure is running in the server?  
And, how can I get this problem worked out?

BTW, in my environment, the SYBASE Database Server also runs on my laptop as well as the VB app does.

Basically – stop using the cursor in your procedure. It's not necessary. You will need to post this in a Sybase group to get the correct syntax, but in SQL Server I would do this:

```
IF EXISTS (SELECT * FROM table
WHERE ...)
UPDATE table ...
ELSE
INSERT table (...
```

—  
Microsoft MVP — ASP/ASP.NET  
Please reply to the newsgroup. The email account listed in my From header is my spam trap, so I don't check it very often. You will get a quicker response by posting to the newsgroup.

—  
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"

Re: Huge memory consumption of ADODB Connection object

---

Microsoft MVP -- ASP/ASP.NET

Please reply to the newsgroup. The email account listed in my From header is my spam trap, so I don't check it very often. You will get a quicker response by posting to the newsgroup.

.