

Re: How does ADO 2 update a SQL stored procedure recordset?

Source: <http://www.tech-archive.net/Archive/Data/microsoft.public.data.ado/2006-01/msg00006.html>

- *From:* "Stephen Howe" <stephenPOINThoweATtns-globalPOINTcom>
 - *Date:* Wed, 4 Jan 2006 14:33:05 -0000
-

> Generally when I want to use ADO 2 to update a recordset based on a MSSQL
> stored procedure I will handle the OnWillChangeRecord event, and
> explicitly
> call a second stored procedure to perform the update.
>
> If I don't handle the OnWillChangeRecord event (and permissions are set
> properly, etc.) ADO will "magically" update the SQL tables for me. How
> does
> this actually work?

Well if you have SELECTed the columns in the recordset that form the Primary Key on the database, ADO has all it needs to update the record, it is unique. If you change the Primary Key in your Recordset I believe ADO is still okay as an ADO Recordset also has the old values of the Primary Key to work with.

You should get an error message if you do not have all the PK columns SELECTed, the Recordset no longer has a unique way of updating the field.

> Suppose a stored procedure joins two or more tables: how does ADO figure
> out which tables are to be updated, when all it would seem ADO has to work
> with is the resultset returned by the stored procedure?

Same thing I believe. I have never used this.

But you do need the PK of each table as part of the SELECTed columns.

Mind you see

<http://msdn.microsoft.com/library/en-us/ado270/htm/mdprodynuniquetable.asp>

> (To clarify, I know that it DOES work, I would just like to understand HOW.)

Well on this that is easy.

You are running SQL Server.

Why not run SQL Profiler and see the underlying SQL commands that ADO issues?

You can track every single SQL command the provider issues.

Also for client-sided recordsets see this article which shows how you can

Re: How does ADO 2 update a SQL stored procedure recordset?

change the criteria that ADO uses to do an Update:

<http://support.microsoft.com/default.aspx?scid=KB;en-us;q190727>

- > A related question: other than the obvious differences in permission requirements and ability to perform additional tasks in the update stored procedure, are there differences between these two approaches? Is one preferred (in terms of security, performance, stability, etc.)?

I prefer never to use a Recordset.

I tend to update the database using a

- (i) SQL UPDATE statement if the same update is being performed in a global fashion with a WHERE clause. A Connection Execute is enough for this..
- (ii) An Adhoc Command object (if the columns affected are the same ones each time) but the parameters vary. For SQL Server, multiple UPDATE statements can be done (reduces network traffic)
- (iii) Individual SQL UPDATE statements if fields being updated vary each time.

I might use a Recordset if I need to examine the fields in detail, do some calculation before altering some of them back. This is if it is too complex for SQL.

Stephen Howe

-
- Prev by Date: [***RE: Using Rich Text formatting with InfoPath***](#)
 - Next by Date: [***connection string and ADO***](#)
 - Previous by thread: [***RE: Using Rich Text formatting with InfoPath***](#)
 - Next by thread: [***connection string and ADO***](#)
 - Index(es):
 - ◆ [***Date***](#)
 - ◆ [***Thread***](#)