

Strange SQL Server CE 2.0 Problem

Source:

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.framework.compactframework/2005-11/msg011>

- *From:* "Nate Kohari" <nkohari@xxxxxxxxxx>
 - *Date:* 28 Nov 2005 07:53:50 -0800
-

(This is a crosspost from microsoft.public.sqlserver.ce since I figured it was applicable to both.)

I am developing a piece of software written in C# that is deployed on a device running Windows CE 4.2. It connects to an Oracle database via a C# web service and loads information onto the handheld. We've recently moved from storing the data in custom data structures to using a SQL Server CE database that is stored on a flash card inside the device.

Overall, it's worked pretty well, but I've run into a strange problem: every time I populate the device's database with records, the first record of each table is inserted with all of its values blank. As far as I can tell, the information is actually incorrect in the database, and not just returned incorrectly via the subsequent SELECT query. Unfortunately, I can't use the SQL CE Query Analyzer -- the device doesn't have a touchscreen, and the Query Analyzer apparently wasn't designed to be navigated with a keyboard only.

I originally was using the SqlDbType types, and switched to DbType types. This sort of fixed the problem -- instead of being entirely blank, the first record's data is just truncated. (The LOCATION_DESC column, which is a 40-character NVARCHAR, is truncated to 6 characters.)

Here's a snippet of the code I'm using to insert/update the records:

```
transaction = _connection.BeginTransaction();

// --- Initialize the SELECT command.

sql = "SELECT COUNT(*) FROM LOCATIONS WHERE LOCATION_CODE = ?";

selectCommand = new SqlCeCommand(sql, _connection, transaction);
selectCommand.Parameters.Add("LOCATION_CODE", DbType.String);
selectCommand.Prepare();

// --- Initialize the INSERT command.
```

Strange SQL Server CE 2.0 Problem

```
sql = "INSERT INTO LOCATIONS (LOCATION_CODE, LOCATION_DESC,  
LOCATION_TYPE, CURRENT_LOCATION) "  
+ "VALUES (?, ?, ?, ?)";
```

```
insertCommand = new SqlCeCommand(sql, _connection, transaction);  
insertCommand.Parameters.Add("LOCATION_CODE", DbType.String);  
insertCommand.Parameters.Add("LOCATION_DESC", DbType.String);  
insertCommand.Parameters.Add("LOCATION_TYPE", DbType.String);  
insertCommand.Parameters.Add("CURRENT_LOCATION", DbType.String);  
insertCommand.Prepare();
```

```
// --- Initialize the UPDATE command.
```

```
sql = "UPDATE LOCATIONS SET LOCATION_DESC = ?, LOCATION_TYPE = ?, "  
+ "CURRENT_LOCATION = ? WHERE LOCATION_CODE = ?";
```

```
updateCommand = new SqlCeCommand(sql, _connection, transaction);  
updateCommand.Parameters.Add("LOCATION_DESC", DbType.String);  
updateCommand.Parameters.Add("LOCATION_TYPE", DbType.String);  
updateCommand.Parameters.Add("CURRENT_LOCATION", DbType.String);  
updateCommand.Parameters.Add("LOCATION_CODE", DbType.String);  
updateCommand.Prepare();
```

```
foreach (Location location in locations)  
{  
    // Determine whether the record exists or not, and insert or update  
    it accordingly.  
    selectCommand.Parameters["LOCATION_CODE"].Value =  
    location.LocationCode;  
    existingRecords = Convert.ToInt32(selectCommand.ExecuteScalar());  
  
    if (existingRecords == 0)  
    {  
        insertCommand.Parameters["LOCATION_CODE"].Value =  
        location.LocationCode;  
        insertCommand.Parameters["LOCATION_DESC"].Value =  
        location.Description;  
        insertCommand.Parameters["LOCATION_TYPE"].Value = location.Type;  
        insertCommand.Parameters["CURRENT_LOCATION"].Value =  
        (location.IsCurrentLocation ? "Y" : " ");  
        insertCommand.ExecuteNonQuery();  
    }  
    else  
    {  
        updateCommand.Parameters["LOCATION_CODE"].Value =  
        location.LocationCode;  
        updateCommand.Parameters["LOCATION_DESC"].Value =  
        location.Description;  
        updateCommand.Parameters["LOCATION_TYPE"].Value = location.Type;  
        updateCommand.Parameters["CURRENT_LOCATION"].Value =  
        (location.IsCurrentLocation ? "Y" : " ");
```

Strange SQL Server CE 2.0 Problem

```
updateCommand.ExecuteNonQuery();  
}
```

```
count++;  
}
```

```
transaction.Commit();
```

Is the problem maybe due to the way I'm running the queries? Since you have to manually collect the SqlCeCommand objects, I'm trying to reuse them as much as possible.

Any help you could provide would be greatly appreciated.

Thanks,
Nate

• ***Follow-Ups:***

◆ ***Re: Strange SQL Server CE 2.0 Problem***

◇ *From:* Ruslan Trifonov

- Prev by Date: ***Re: Help with deciding on best solution***
- Next by Date: ***Re: Arghhh!! How to do Resolution aware app VS2005***
- Previous by thread: ***Re: Help with deciding on best solution***
- Next by thread: ***Re: Strange SQL Server CE 2.0 Problem***
- Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***