An End to "String or binary data would be truncated"

Posted At : November 21, 2005 5:49 PM | Posted By : Steve
Related Categories: DataMgr

If you use MS SQL Server, you have seen this error. It is annoying enough that you are getting an error, but this is one that doesn't tell you which field is causing the problem - leaving you to guess for yourself.

The first and most obvious solution is the use of the maxlength attribute of <cfqueryparam>. This is helpful in as much as the error that you get comes from the cfqueryparam tag instead of from the database - making it easier to determine which field has the problem.

The simplest way to prevent the error is to wrap a Left() function around the incoming data for that field, to truncate it to the length of the allowable data.

The main problem with these solutions is that you have to specify the length and if you decide to change the length of the field in the database, you have to remember to change your code as well.

Another option is to use my free **DataMgr component** for your inserts and updates. The DataMgr will actually tell you what field the error is on without you needing to specify the field length.

A newly added feature to DataMgr will help prevent the error altogether. The new truncate() method will return a structure of data truncated to the allowable length of each given field. This will allow you to insert or update a record using that data knowing that you won't get an error for data that is too long.

To use truncate() simply pass in the name of the table and the structure that you want to truncate and you will get back that same structure, truncated to the length of the fields matching the keys of the structure (this is the same sort of structure used for the insertRecord and updateRecord methods of DataMgr).

So, if you wanted to insert a record to a table named "mytable" using a structure named "mydata" and you wanted to circumvent an errors for text that is too long, you could do this:
<cfset DataMgr.insertRecord('mytable',DataMgr.truncate('mytable',mydata))>

Regardless of what solution you use on the server-side, the maxlength attribute of the input field will help on the client-side. This will prevent the user from entering more text into the text field than your database field can handle. This doesn't obviate the need for a solution on the server-side, but does make the applicaion more user-friendly.

Again, you must change this number if you change the length of your field in the database. I don't have a solution for this (yet), but I hope to add this capability to my free **sebForm custom tags** soon (well, eventually).

Whether or not you decide to use the DataMgr component, this will hopefully help you handle this common database error.