Cfqueryparam Fails When Using "WHERE EXISTS"

Posted At : December 6, 2006 2:10 PM | Posted By : Mark Kruger
Related Categories: MS SQL Server, Coldfusion & Databases

I have always been an advocate of *Cfqueryparam*. Binding your variables innoculates you against SQL injection attack, often results in speed improvements and lessens the load on your database server. It may even help with the laundry (that's the word on the street). This morning I ran across an error that is produced by the *correct* use of cfquery param. It has to do with using the clause "WHERE EXISTS" in your query. Here's the query in question.

WHERE EXISTS Sample

```
<cfquery name="test" datasource="mydsn" result="r">
        SELECT     h.column1
        FROM     r_tableA h
        WHERE     EXISTS (    select     1
                        FROM     R_tableA_tableB hf,
                                R_TableC f
                        WHERE    hf.col_id = f.col_id
                                and f.columnBlah = <cfqueryparam cfsqltype="CF_SQL_CHAR"
value="var1"/>
                                and f.colblah2 = <cfqueryparam cfsqltype="CF_SQL_CHAR"
value="var2"/>
                                and hf.col2_id = h.col2_id
                        )
        order by h.blah
    </cfquery>
```

This query is not mine and I don't generally use this syntax. Perhaps a read can inform me as to when it is appropriate to use the "EXISTS" in this way. In any case the query resulted in an error - [Macromedia][SQLServer JDBC Driver][SQLServer]Could not find prepared statement with handle 1. Because of the nature of the error I surmised pretty quickly that the binding was not working (paramter 1 meaning the first declared parameter in the prepare statment). Removing Cfqueryparam fixed the error - but I wanted them back. So I rewrote the query as:

```
<cfquery name="test" datasource="mydsn" result="r">
         DECLARE @paramA char(25)
        DECLARE @paramB    char(25)

        SELECT @paramA = <cfqueryparam cfsqltype="CF_SQL_CHAR" value="var1"/>
        SELECT @paramB = <cfqueryparam cfsqltype="CF_SQL_CHAR" value="var2"/>

        SELECT     h.column1
        FROM     r_tableA h
        WHERE     EXISTS (    select     1
                        FROM     R_tableA_tableB hf,
                                R_TableC f
                        WHERE    hf.col_id = f.col_id
                                and f.columnBlah = @paramA
                                and f.colblah2 = @paramB
                                and hf.col2_id = h.col2_id
                        )
        order by h.blah
    </cfquery>
```

Executing the binding outside the nested "exists" query allowed the variables to be declared and used correctly. As far as *why* this error occurs I can only surmise that the parameter declaration happens "inside" the nested query but is accessed "outside" somehow in the prepared statement. In other words the SET statement is coming before the declaration. I'm not sure why that would be the case.