

Changing Database Schemas may require a restart of CFMX

Posted At : April 29, 2005 4:01 PM | Posted By : Mark Kruger

Related Categories: SQL tips, Coldfusion & Databases

There's a tricky nuance that you must take into account when you make schema changes in MS SQL. When we discovered the following behavior I looked to see if I could find it blogged or documented somewhere. Failing that, I thought someone else out there might benefit from hearing about this issue. You might run into problems if you are dealing with the following conditions:

- CFMX using JDBC drivers to MS SQL
- A view with a "select *" in it for one or more tables
- The need to change the schema of a particular table referenced with the asterisk (*) within the view

Here's what happens (and how to fix it).

h4>How we discovered it

We first found out about this bug when we made a minor modification to an application that was built by a third party. The application had a view created with code similar to this:

```
CREATE VIEW dbo.vwItemFeatures
AS
SELECT      L.*,
            LF.FONT,
            LF.ICON,
            LF.CATDISPLAY,
            LF.FEAT
FROM ITEM_LIST L
LEFT JOIN
    ITEM_FEATURE LF ON L.ITEM_ID = LF.ITEM_ID
```

This view was used to select featured items in an e-commerce type applications. We needed to add a field to the *ITEM_LIST* table. After we added the field we got a variety of similar errors for queries accessing the view - queries that had previously worked. All of the errors were type binding errors. The JDBC driver was complaining that the *type* being bound to the column via `<cfqueryparam ...>` did not match the type of the actual column in question.

For example, we had the following query:

```
SELECT      *
FROM      vwItemFeatures
WHERE      description LIKE
(<cfqueryparam cfsqltype="CF_SQL_CHAR" value="#likeformat(keyword)#">)
```

The JDBC error we got back was *"Syntax error converting the varchar value 'Widgets' to a column of data type int"*. That's odd because we were binding a char value - right?

The reason

It turns out that 2 things are going on.

1. **SQL Server uses the *old* schema** - If add the column to the table, then do a quick

"select *" from the view you will see that the new column *is not returned*. SQL caches or stores the schema for the table at the time it was made and does not refresh it (at least not immediately).

2. **JDBC requests use the *new* schema** - Somehow, JDBC gets wind of the new schema in the table and builds its column list based on the new order. Internally, some sort of ordering or id'ing of columns by their position is going on. So the JDBC driver passing the column name in the where clause ends up referencing a column of a different type - in this case, the JDBC driver was attempting to pass character data to a column of an integer type.

This is a tricky issue. Think about the implications. What if your view table is mostly character data and adding a column *doesn't* throw an error. You could end up breaking your search or viewing entirely wrong data. For example, what if you passed in "lastname" in a search and the re-ordering caused the search to be by City instead? You could end up with folks from Johnsonville instead of folks named Johnson - even though the query clearly showed

```
WHERE lastname LIKE ('%Johnson%').
```

Talk about a head scratcher. I might try to test that out and see if it's possible.

The fix

Restart CF server, right? Nope - you have to get both sides synchronized. First, refresh the view. I do this in query analyzer using the "alter view" code (right click on the view and choose "script to new window as alter"). Simple run the alter view without changing the syntax and it will refresh the schema for you. Then you will need to restart CF. I know that doesn't make sense. It would seem like getting them both on the same schema by refreshing the view would do the trick. I can only surmise that refreshing the view changes the ordering yet again - perhaps based on indexing and execution plan. If you have heard of this issue or have any correction or insight into my suppositions let me know.