Custom fields displaying as Chinese / Korean characters after upgrade

Last Modified on 09/24/2024 10:38 am EDT

Symptom

Custom fields on OnGuard forms are now displaying as Chinese or Korean characters after an upgrade.

After an OnGuard upgrade, certain custom fields that were originally generated with VARCHAR datatype are not converted to NVARCHAR type (Unicode), which causes these fields to display in Chinese or Korean characters in System Administration, but appear normally in the database as English characters.

Resolution

1. Create Database backup.
2. In SQL Management Studio ,navigate to Tools>Options>Designers> and uncheck "Prevent saving changes that require table re-creation".
   Note: This feature should be turned back on once complete
3. Get all character tables without NVARCHAR data type from the AccessControl database. This can be done by executing the following query:

   ```
   SELECT * FROM INFORMATION_SCHEMA.COLUMNS WHERE DATA_TYPE = 'VARCHAR';
   ```

   ```
   SELECT * FROM INFORMATION_SCHEMA.COLUMNS WHERE DATA_TYPE = 'CHAR';
   ```

   In this case we returned 4 tables; normally we would expect 0. Of importance is the Table Name, Column, and Character Max Length that contains the VARCHAR data type.

   ```
   TABLE COLUMN CHARACTER_MAXIMUM_LENGTH
   LST720 NAME 34
   Department NAME 60
   CITY NAME 32
   NET_EMP_SID USER_SID 64
   ```
4. Prepare Alter Table Statement. You want to convert the data type on these tables from VARCHAR to NVARCHAR. Use the ALTER TABLE in the following way:

   ```
   ALTER TABLE [LST720] ALTER COLUMN [Name] NVARCHAR(34);
   ALTER TABLE [Department] ALTER COLUMN [Name] NVARCHAR(60);
   ```

   ```
   ALTER TABLE [CITY] ALTER COLUMN [Name] NVARCHAR(32);
   ```

   ```
   ALTER TABLE [NET_EMP_SID] ALTER COLUMN [USER_SID] NVARCHAR(64);
   ```
5. If the Query executes successfully, skip to Step 6. If it fails, you will see a message similar to:

`The object 'xxxx' is dependent on column 'NAME'.`

`Msg 4922, Level 16, State 9, Line 3`

`ALTER TABLE ALTER COLUMN NAME failed because one or more objects access this column.`

To work around this we can manually remove the index on the table then add it back once we run the ALTER TABLE statement:

   a. In SQL Management Studio navigate to the table in question in Object Explorer.
   b. Expand the table, Expand Indexes.
   c. Find the object 'xxxx' name from above in the indexes.
   d. Right click on the Index > Script Index as > Create to New Query Editor Window
   e. Right click on the Index in question and delete > OK.
   f. Run the Alter Table Statement to change the table data type to NVCHAR.
   g. Execute the script generated in step 4 to restore the index.

6.

Note: If the column you are changing the data type on is a primary key, you must script a primary key restraint, delete the primary key, make the change to the table column, then restore the PK restraint.

7. Re-run these queries to make sure all the changes have been correctly affected:

`SELECT * FROM INFORMATION_SCHEMA.COLUMNS WHERE DATA_TYPE = 'VARCHAR';`

`SELECT * FROM INFORMATION_SCHEMA.COLUMNS WHERE DATA_TYPE = 'CHAR';`

Zero rows should be returned. Log out and back into System Administration to verify the fields are correctly displayed.

Applies To

OnGuard (All versions)

Additional Information