The SBO_SP_TransactionNotification Problem

This document explains the problems you can experience due to misconfigurations of the SAP provided Stored Procedure SBO_SP_TransactionNotification

Table of Content

What is the SBO_SP_TransactionNotification?

The SP_TransactionNotification (and SBO_SP_PostTransActionNotice) is to receive notification on any transaction – in other words to do some validation in SAP Business One and prevent the user from doing an action if the validation condition fails. This stored procedure(SP) is created automatically with each database. This stored procedure is fired whenever operations are performed on business objects like document or master records and User-Defined Objects.

Read more here: https://blogs.sap.com/2015/01/27/sbosptransactionnotification/

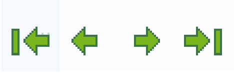
What type of issues can occur?

There are two types of problems that can be caused by these SAP Stored Procedures

1. **SAP Issues:** Various issues that have been introduced due to SAP Bugs in the evolution of the different SAP versions (changes in data-structure and data-lengths)
2. **Reseller Issues:** Issues where a consultant or add-on use the customization options of the stored procedures incorrectly. These are in many cases due to type 1 problems and lack of knowledge thereof.

How do these errors manifest themselves in a Boyum Add-on?

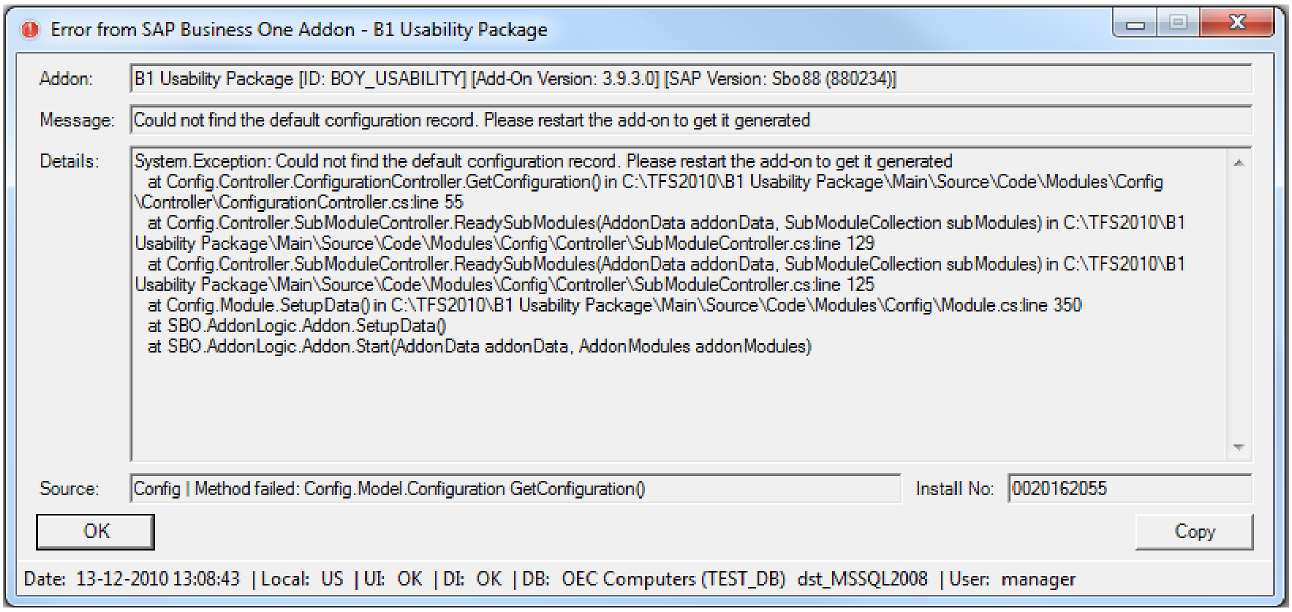
Most issues if SBO_SP_Transaction have problem manifest themselves during startup of the add-on itself of in updating one of its Configurations (more specifically one of the configurations that you can navigate with the arrow keys)



Below are samples of how the issues are shown to the user

Sample issue 1: Main window configuration problem

You get the following error during startup:



Addon: B1 Usability Package [ID: BOY_USABILITY] [Add-On Version: 3.9.3.0] [SAP Version: Sbo88 (880234)]

Message: Could not find the default configuration record. Please restart the add-on to get it generated

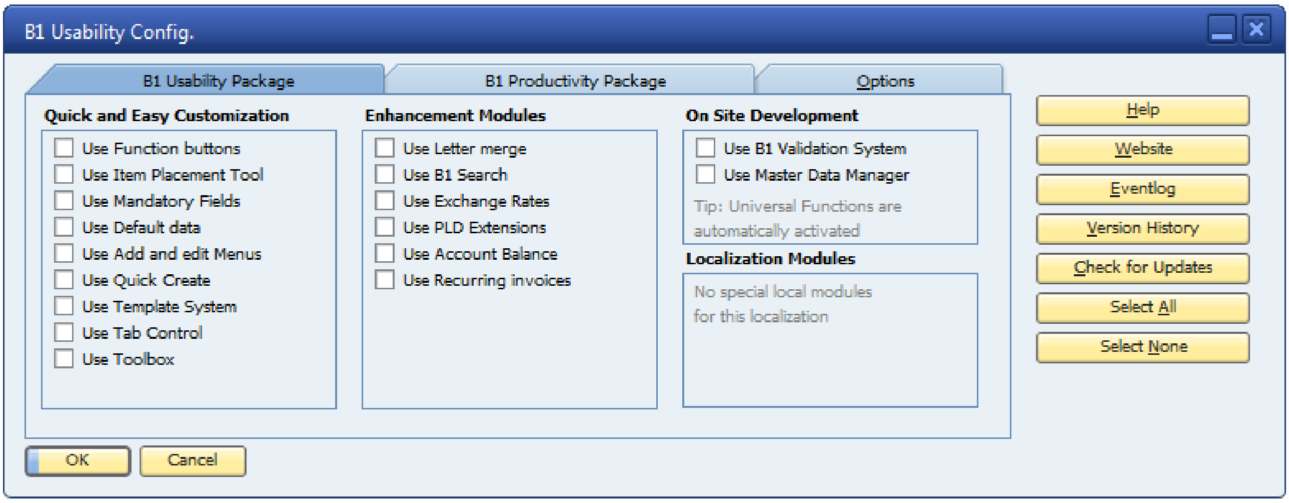
Details:

```
System.Exception: Could not find the default configuration record. Please restart the add-on to get it generated
   at Config.Controller.ConfigurationController.GetConfiguration() in C:\TFS2010\B1 Usability Package\Main\Source\Code\Modules\Config
\Controller\ConfigurationController.cs:line 55
   at Config.Controller.SubModuleController.ReadySubModules(AddonData addonData, SubModuleCollection subModules) in C:\TFS2010\B1
Usability Package\Main\Source\Code\Modules\Config\Controller\SubModuleController.cs:line 129
   at Config.Controller.SubModuleController.ReadySubModules(AddonData addonData, SubModuleCollection subModules) in C:\TFS2010\B1
Usability Package\Main\Source\Code\Modules\Config\Controller\SubModuleController.cs:line 125
   at Config.Module.SetupData() in C:\TFS2010\B1 Usability Package\Main\Source\Code\Modules\Config\Module.cs:line 350
   at SBO.AddonLogic.Addon.SetupData()
   at SBO.AddonLogic.Addon.Start(AddonData addonData, AddonModules addonModules)
```

Source: Config | Method failed: Config.Model.Configuration GetConfiguration() Install No: 0020162055

Date: 13-12-2010 13:08:43 | Local: US | UI: OK | DI: OK | DB: OEC Computers (TEST_DB) dst_MSSQL2008 | User: manager

Or you are able to show the B1UP main configuration, but unable to check any of the checkboxes



Quick and Easy Customization

□ Use Function buttons
□ Use Item Placement Tool
□ Use Mandatory Fields
□ Use Default data
□ Use Add and edit Menus
□ Use Quick Create
□ Use Template System
□ Use Tab Control
□ Use Toolbox

Enhancement Modules

□ Use Letter merge
□ Use B1 Search
□ Use Exchange Rates
□ Use PLD Extensions
□ Use Account Balance
□ Use Recurring invoices

On Site Development

□ Use B1 Validation System
□ Use Master Data Manager

Tip: Universal Functions are automatically activated

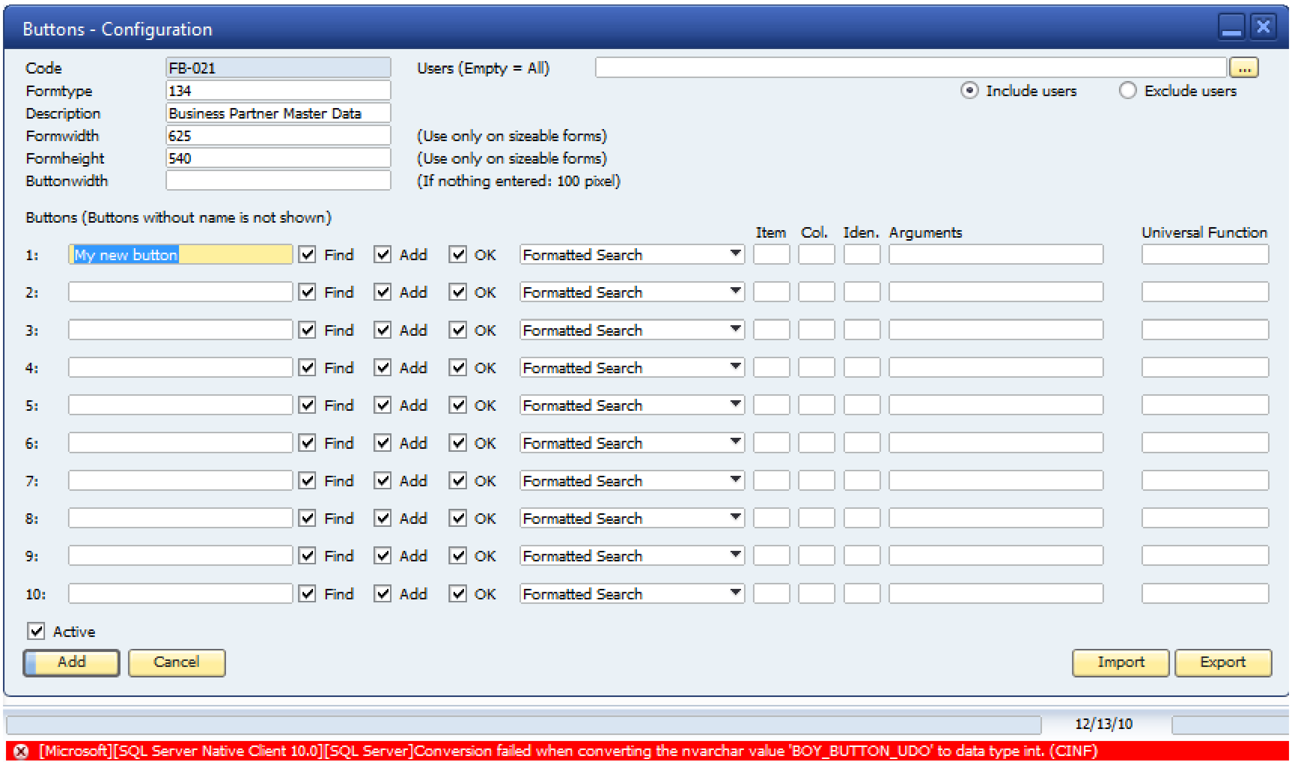
Localization Modules

No special local modules for this localization

Sample issue 2: Configuration add problem

When trying to add, update or remove a configuration in B1 Usability Package (Or any other add-on using the technique called User Defined Object):

[Microsoft][SQL Native Client][SQL Server]Error converting data type nvarchar to int.(CINF)



Code FB-021

Formtype 134

Description Business Partner Master Data

Formwidth 625 (Use only on sizeable forms)

Formheight 540 (Use only on sizeable forms)

Buttonwidth (If nothing entered: 100 pixel)

Users (Empty = All)

● Include users ○ Exclude users

Buttons (Buttons without name is not shown)

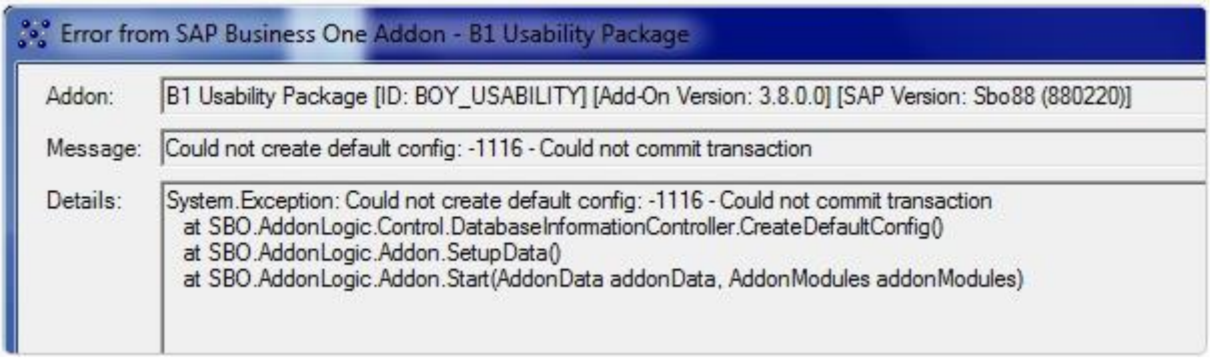
						Item	Col.	Iden.	Arguments	Universal Function
1:	My new button	☑ Find	☑ Add	☑ OK	Formatted Search					
2:		☑ Find	☑ Add	☑ OK	Formatted Search					
3:		☑ Find	☑ Add	☑ OK	Formatted Search					
4:		☑ Find	☑ Add	☑ OK	Formatted Search					
5:		☑ Find	☑ Add	☑ OK	Formatted Search					
6:		☑ Find	☑ Add	☑ OK	Formatted Search					
7:		☑ Find	☑ Add	☑ OK	Formatted Search					
8:		☑ Find	☑ Add	☑ OK	Formatted Search					
9:		☑ Find	☑ Add	☑ OK	Formatted Search					
10:		☑ Find	☑ Add	☑ OK	Formatted Search					

☑ Active

12/13/10

[Microsoft][SQL Server Native Client 10.0][SQL Server]Conversion failed when converting the nvarchar value 'BOY_BUTTON_UDO' to data type int. (CINF)

Sample issue 3: You get the following error during startup



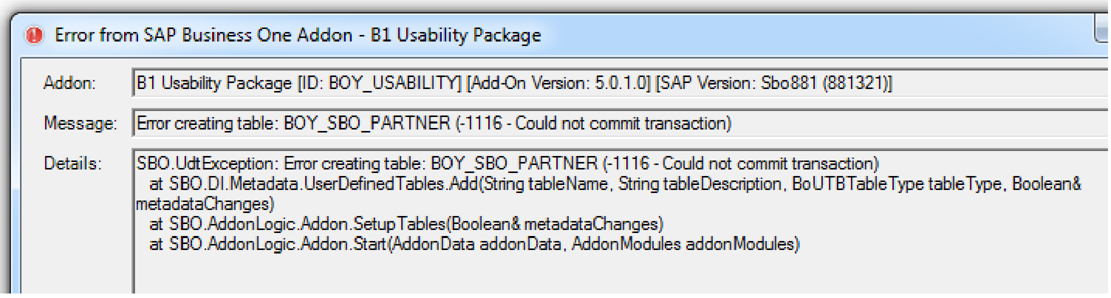
Addon: B1 Usability Package [ID: BOY_USABILITY] [Add-On Version: 3.8.0.0] [SAP Version: Sbo88 (880220)]

Message: Could not create default config: -1116 - Could not commit transaction

Details:

```
System.Exception: Could not create default config: -1116 - Could not commit transaction
  at SBO.AddonLogic.Control.DatabaseInformationController.CreateDefaultConfig()
  at SBO.AddonLogic.Addon.SetupData()
  at SBO.AddonLogic.Addon.Start(AddonData addonData, AddonModules addonModules)
```

Sample issue 4: You get the following error during startup



Error from SAP Business One Addon - B1 Usability Package

Addon: B1 Usability Package [ID: BOY_USABILITY] [Add-On Version: 5.0.1.0] [SAP Version: Sbo881 (881321)]

Message: Error creating table: BOY_SBO_PARTNER (-1116 - Could not commit transaction)

Details:

```
SBO.UdtException: Error creating table: BOY_SBO_PARTNER (-1116 - Could not commit transaction)
   at SBO.DI.Metadata.UserDefinedTables.Add(String tableName, String tableDescription, BoUTBTableType tableType, Boolean&
metadataChanges)
   at SBO.AddonLogic.Addon.SetupTables(Boolean& metadataChanges)
   at SBO.AddonLogic.Addon.Start(AddonData addonData, AddonModules addonModules)
```

(The table mentioned in the error might be different, and it can also refer to an issue adding a UDF, KEY or UDO, but since tables are created first in add-ons it is most likely to see the issue with a table)

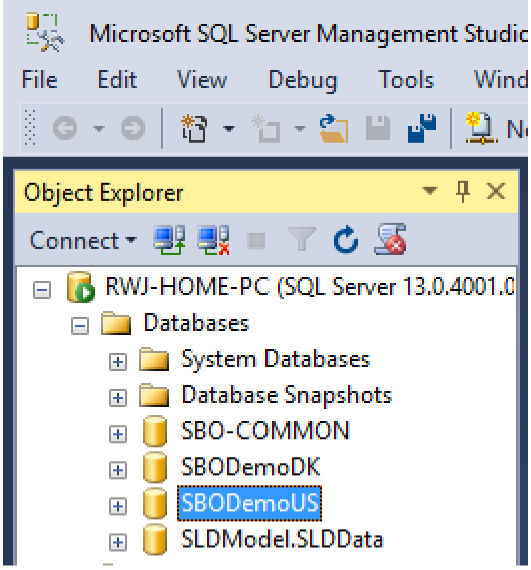
Sample issue 5: You get an error message that indicate an error with some SQL but error message "aka empty brackets ()" [HANA only]

This issue is caused by the fact that the Store Procedures on HANA have changed it's parameter "Object_type" to have length NVARCHAR(40) instead of the old NVARCHAR(20)

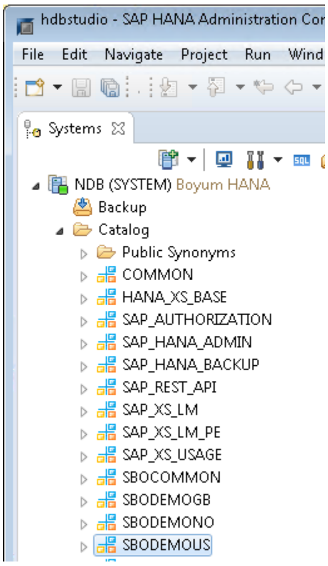
Read more about this specific issue here: www.boyum-it.com/Link/HANA_SP_ISSUE

How to get to place of the issue?

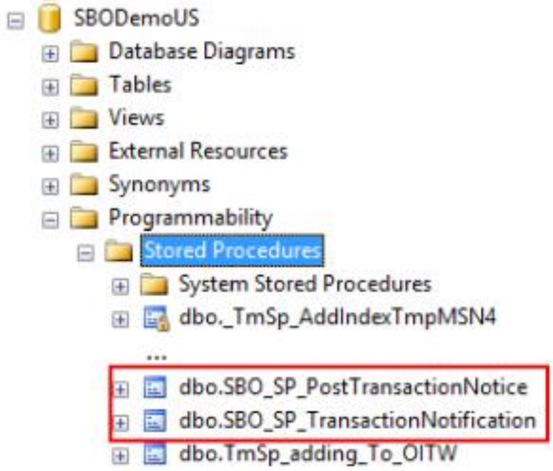
<table>
<tr><th>MSSQL</th><th>HANA</th></tr>
<tr>
<td>On your SAP server go to the SQL Server Management Studio



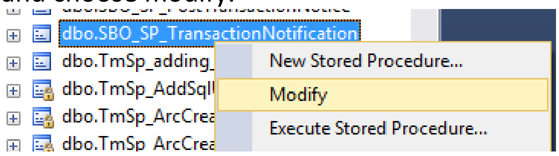
Find the database in question and navigate to the Programmability > Stored Procedures Folder</td>
<td>Go to SAP HANA Studio



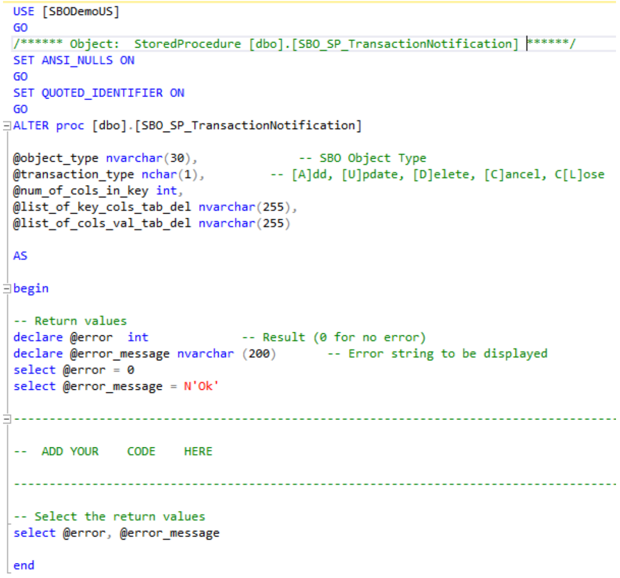
Find the database in question and navigate to the Procedures Folder</td>
</tr>
</table>



Find the stored procedure called dbo.SBO_SP_TransactionNotification; right click it and choose modify.

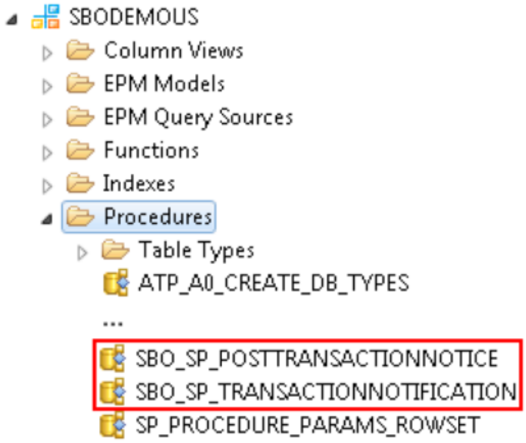


This will open up the code to the stored procedure that holds the problem. The following screenshot show the code without any modifications (SBO_SP_PostTransactionNotice similar in structure).

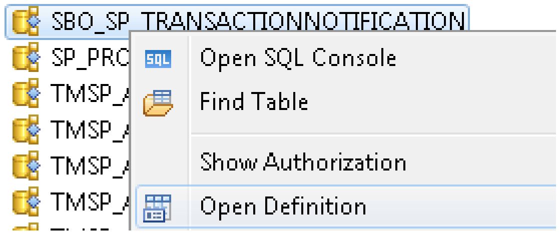


```
USE [SBODemoUS]
GO
/****** Object:  StoredProcedure [dbo].[SBO_SP_TransactionNotification] ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER proc [dbo].[SBO_SP_TransactionNotification]

@object_type nvarchar(30),              -- SBO Object Type
@transaction_type nchar(1),          -- [A]dd, [U]pdate, [D]elete, [C]ancel, C[L]ose
@num_of_cols_in_key int,
@list_of_key_cols_tab_del nvarchar(255),
@list_of_cols_val_tab_del nvarchar(255)

AS

begin

-- Return values
declare @error  int              -- Result (0 for no error)
declare @error_message nvarchar (200)       -- Error string to be displayed
select @error = 0
select @error_message = N'Ok'

--------------------------------------------------------------------------------

--  ADD YOUR    CODE    HERE

--------------------------------------------------------------------------------

-- Select the return values
select @error, @error_message

end
```

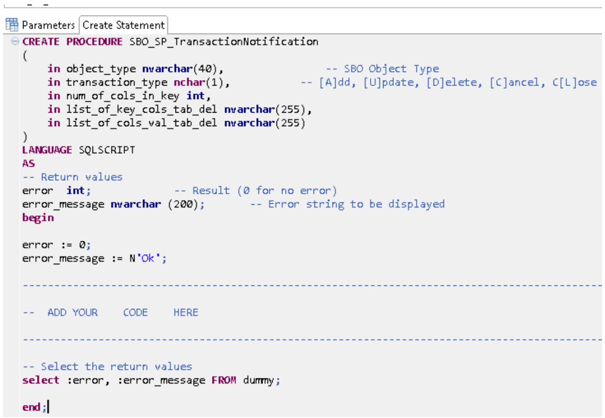
NB: AS THIS IS A LEGAL STORED PROCEDURE TO MODIFY THERE MIGHT BE SOMETHING IN THE ADD YOUR CODE HERE SECTION DEPENDING ON THE OTHER ADD-ONS YOU ARE RUNNING.



Find the stored procedure called SBO_SP_TRANSACTIONNOTIFICATION, right-click it and choose “Open Defintion”



This will open up the code to the stored procedure that holds the problem. The following screenshot show the code without any modifications (SBO_SP_PostTransactionNotice similar in structure).



```
CREATE PROCEDURE SBO_SP_TransactionNotification
(
    in object_type nvarchar(40),                -- SBO Object Type
    in transaction_type nchar(1),          -- [A]dd, [U]pdate, [D]elete, [C]ancel, C[L]ose
    in num_of_cols_in_key int,
    in list_of_key_cols_tab_del nvarchar(255),
    in list_of_cols_val_tab_del nvarchar(255)
)
LANGUAGE SQLSCRIPT
AS
-- Return values
error  int;             -- Result (0 for no error)
error_message nvarchar (200);       -- Error string to be displayed
begin

error := 0;
error_message := N'Ok';

--------------------------------------------------------------------------------

--  ADD YOUR    CODE    HERE

--------------------------------------------------------------------------------

-- Select the return values
select :error, :error_message FROM dummy;

end;
```

NB: AS THIS IS A LEGAL STORED PROCEDURE TO MODIFY THERE MIGHT BE SOMETHING IN THE ADD YOUR CODE HERE SECTION DEPENDING ON THE OTHER ADD-ONS YOU ARE RUNNING.

NB: YOU CAN’T MODIFY A STORED PROCEDURE ON HANA, SO IN ORDER TO MODIFY IT YOU NEED TO DROP THE PROCEDURE AND RE-CREATE IT AGAIN WITH THE

	MODIFICATION. FOR THAT REASON IT IS NOT RECOMMENDED THAT IT IS DONE WHILE PEOPLE USE THE DATABASE!

How to determine what the problem is and fix it

There can be one of two problems wrong with the code (or in some cases both)

NB: BOYUM IT A/S CANNOT TAKE ANY RESPONSIBILITY IN THE CHANGES YOU MAKE IN THE SBO_SP_TRANSACTIONNOTIFICATION SINCE EACH CASE IS UNIQUE, AND YOU SHOULD ALWAYS MAKE A BACKUP OF THE ORIGINAL BEFORE PROCEEDING.

1. Incorrect use of the SBO_SP_TransactionNotification by other add-ons/consultants

This can only happen in the case where you find code in the ADD YOU CODE HERE section. Some add-ons add code here to implement their features and also some consultants use it for custom modification.

IF YOU KNOW WHO ARE RESPONSIBLE FOR THE CUSTOM CODE, YOU SHOULD CONTACT THEM AND CONFRONT THEM WITH THE ISSUE. ELSE YOU CAN SAVE AND SEND A COPY OF YOUR CODE OR TAKE A SCREENSHOT TO BOYUM IT SUPPOT AT support@boyum-it.com AND WE WILL TAKE A LOOK AT IT AND ADVICE YOU ON THE MISCONFIGURATION.

NB: THE SAMPLES HERE IS SHOWN IN MSSQL T-SQL BUT CONCEPTS AND SOLUTIONS ARE THE SAME ON HANA

Sample 1: Numeric vs Alphanumeric values

This is a sample of a SBO_SP_transactionNotification that have been modified incorrectly by a consultant that incorrectly misunderstood the type of the @object_type variable.

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER proc [dbo].[SBO_SP_TransactionNotification]
@object_type nvarchar(20),                  -- SBO Object Type
@transaction_type nchar(1),             -- [A]dd, [U]pdate, [D]elete, [C]ancel, C[L]ose
@num_of_cols_in_key int,
@list_of_key_cols_tab_del nvarchar(255),
@list_of_cols_val_tab_del nvarchar(255)
AS
begin
-- Return values
declare @error  int                 -- Result (0 for no error)
declare @error_message nvarchar (200)       -- Error string to be displayed
select @error = 0
select @error_message = N'Ok'
--------------------------------------------------------------------------------------

IF(@object_type = 13)     ←     Issue
BEGIN
    --Do stuff on A/R Invoice
END

IF(@object_type = 17)     ←     Issue
BEGIN
    --Do stuff on Sales Order
END

--------------------------------------------------------------------------------------
-- Select the return values
select @error, @error_message
end
```

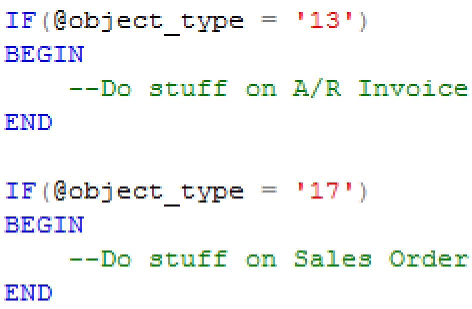
As you can see, there have been added a few SQL-IF’s that check if the @object_type is of type A/R Invoice ('13') or if the object_type is a sales order ('17'), but the consultant expected that all values of @object_type was numeric (a number), but it is not for Userdefined Objects and User-defiend tables (Indicated by the fact that object_type is of type nvarchar(20) and not INT.

How to fix it

The problem here is that @object_type is assumed numeric and not alphanumeric. 13 and 17 are inserted as numeric values and therefore a conversion will take place every time the line is executed. This automatic conversion will work for types like A/R invoice (which @object_type is '13') but for custom object_types like B1UP - Function Buttons (which @object_type is 'BOY_BUTTONS') this conversion will fail.

The fix how-ever is quite easy. All you need to do is assure that all checks are done on an alphanumeric level.

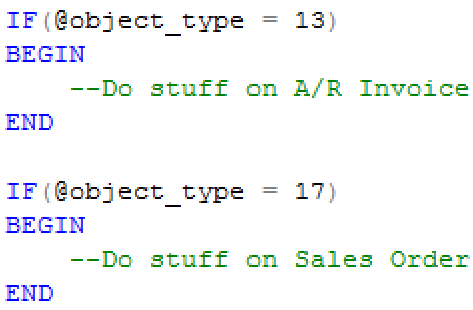
Correct



```
IF(@object_type = '13')
BEGIN
    --Do stuff on A/R Invoice
END

IF(@object_type = '17')
BEGIN
    --Do stuff on Sales Order
END
```

All we do is add ‘ ’ around the values

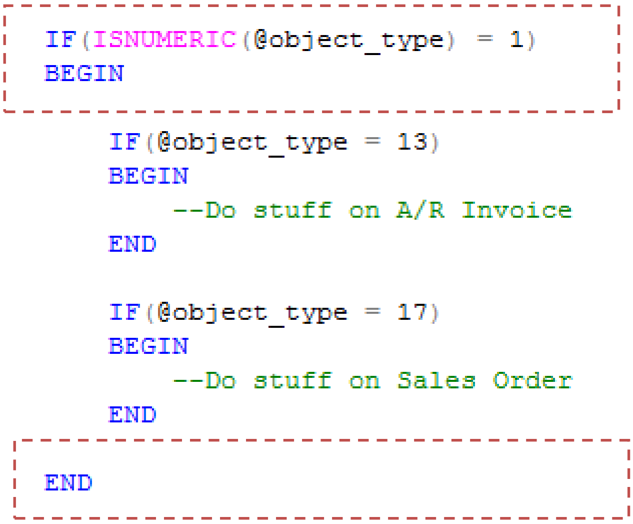
Incorrect



```
IF(@object_type = 13)
BEGIN
    --Do stuff on A/R Invoice
END

IF(@object_type = 17)
BEGIN
    --Do stuff on Sales Order
END
```

If you find it hard to find all instances in a very long modification you can alternative do the following

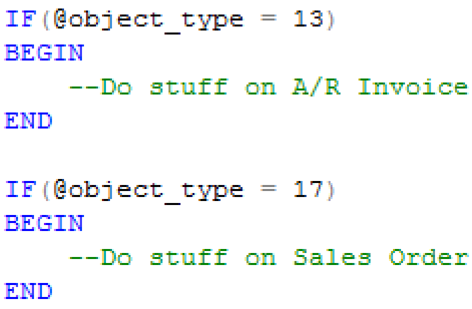
Correct



```
IF(ISNUMERIC(@object_type) = 1)
BEGIN

    IF(@object_type = 13)
    BEGIN
        --Do stuff on A/R Invoice
    END

    IF(@object_type = 17)
    BEGIN
        --Do stuff on Sales Order
    END

END
```

Here we surround the ‘Incorrect’ code to only run if the object_type in fact is numeric

Incorrect

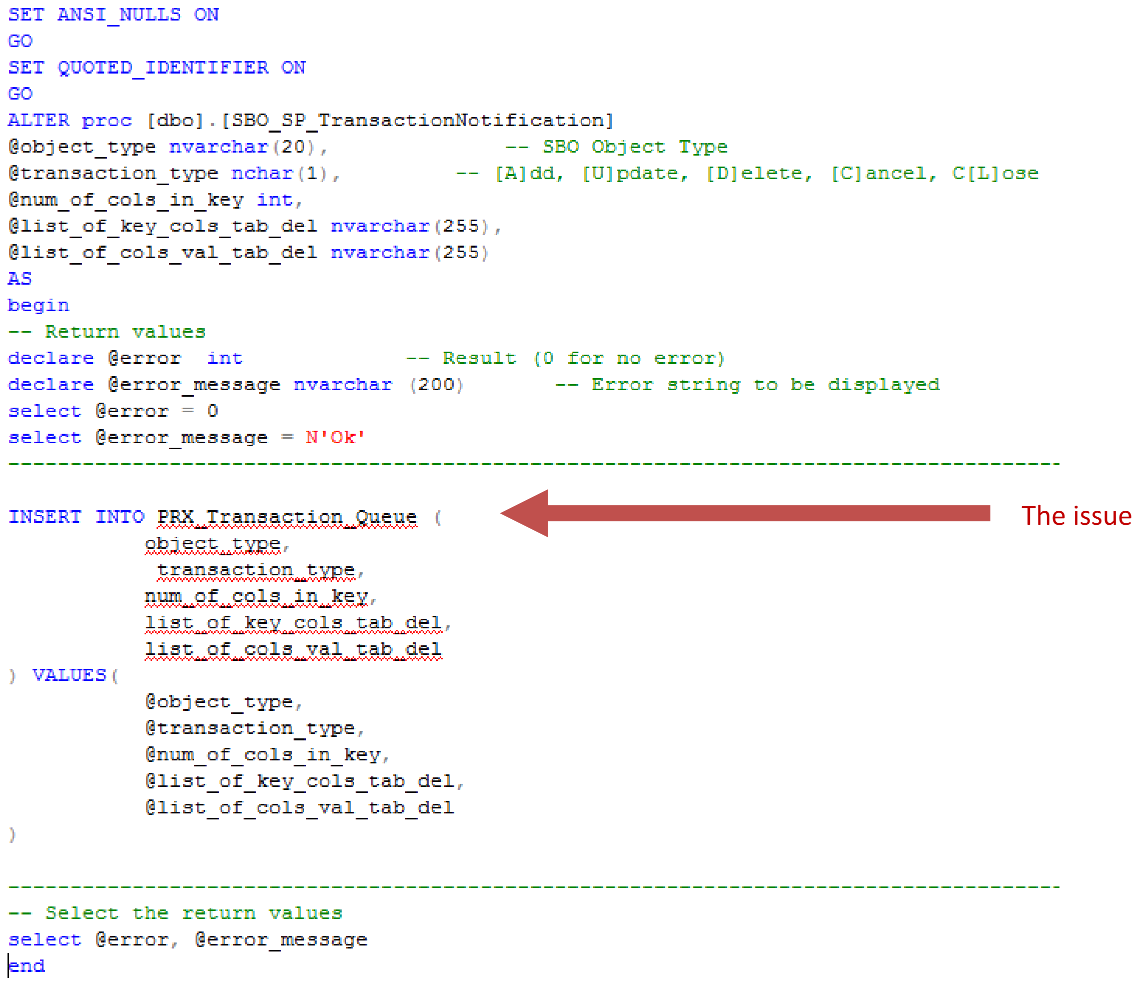


```
IF(@object_type = 13)
BEGIN
    --Do stuff on A/R Invoice
END

IF(@object_type = 17)
BEGIN
    --Do stuff on Sales Order
END
```

NB: ISNUMERIC DOES NOT EXIST ON HANA SO SOMETHING LIKE THIS NEED TO BE DONE IS YOU NEED IT: HTTP://STACKOVERFLOW.COM/QUESTIONS/27703567/WHAT-IS-THE-EQUIVALENT-OF-T-SQL-ISNUMERIC-FUNCTION-IN-HANA-SQLSCRIPT

Sample 2: Inserts into tables/Store procedures with wrong structure

This is a sample of a SBO_SP_transactionNotification that have been modified incorrectly by another add-on (In this case a Praxis Add-on).



```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER proc [dbo].[SBO_SP_TransactionNotification]
@object_type nvarchar(20),               -- SBO Object Type
@transaction_type nchar(1),          -- [A]dd, [U]pdate, [D]elete, [C]ancel, C[L]ose
@num_of_cols_in_key int,
@list_of_key_cols_tab_del nvarchar(255),
@list_of_cols_val_tab_del nvarchar(255)
AS
begin
-- Return values
declare @error  int              -- Result (0 for no error)
declare @error_message nvarchar (200)       -- Error string to be displayed
select @error = 0
select @error_message = N'Ok'
--------------------------------------------------------------------------------------

INSERT INTO PRX_Transaction_Queue (
           object_type,
            transaction_type,
           num_of_cols_in_key,
           list_of_key_cols_tab_del,
           list_of_cols_val_tab_del
) VALUES(
           @object_type,
           @transaction_type,
           @num_of_cols_in_key,
           @list_of_key_cols_tab_del,
           @list_of_cols_val_tab_del
)

--------------------------------------------------------------------------------------
-- Select the return values
select @error, @error_message
end
```

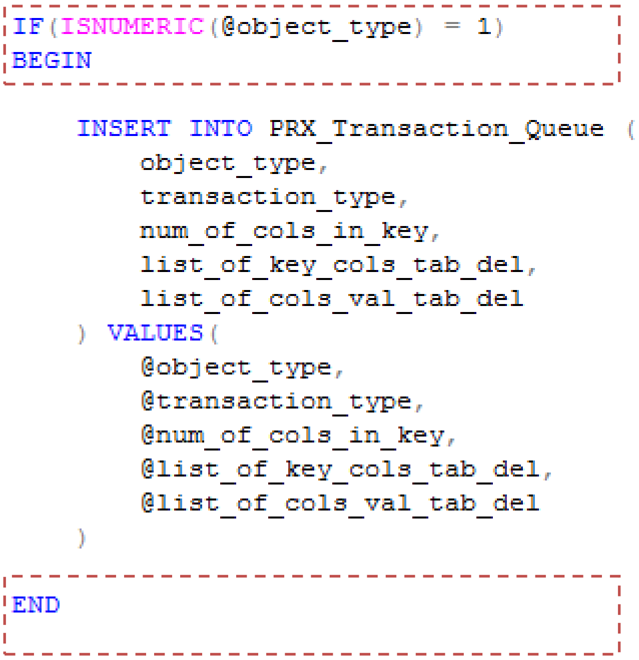
The issue

As you can see, there have been added an extra INSERT SQL statement and that is the issue since it is not able to accept the data given to it.

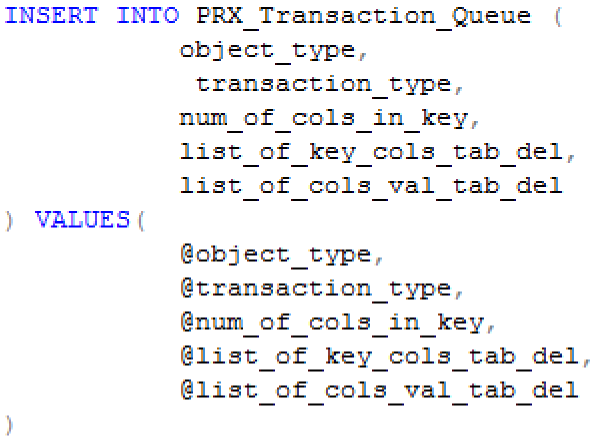
It can in some cases also be a call to another Stored Procedure. In such a case you need to check the code of those stored procedures and make sure it accept the correct length of parameter object_type (NVARCHAR(20) on MSSQL and NVARCHAR(40) on HANA) and this procedure does not contain any of the issues explained in this document.

How to fix it

Since the INSERT SQL here was not expecting alphanumeric we need to make sure that it is only executed when the @object_type is numeric.

<table>
<tr><th>Correct</th><th>Incorrect</th></tr>
<tr><td>
<pre>
IF(ISNUMERIC(@object_type) = 1)
BEGIN

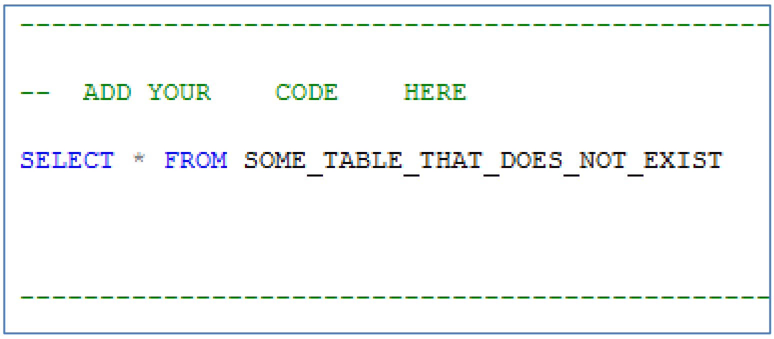
    INSERT INTO PRX_Transaction_Queue (
        object_type,
        transaction_type,
        num_of_cols_in_key,
        list_of_key_cols_tab_del,
        list_of_cols_val_tab_del
) VALUES(
        @object_type,
        @transaction_type,
        @num_of_cols_in_key,
        @list_of_key_cols_tab_del,
        @list_of_cols_val_tab_del
)

END
</pre>
<i>Here we surround the 'Incorrect' code to only run if the object_type in fact is numeric</i></td><td>
<pre>
INSERT INTO PRX_Transaction_Queue (
            object_type,
             transaction_type,
            num_of_cols_in_key,
            list_of_key_cols_tab_del,
            list_of_cols_val_tab_del
) VALUES(
            @object_type,
            @transaction_type,
            @num_of_cols_in_key,
            @list_of_key_cols_tab_del,
            @list_of_cols_val_tab_del
)
</pre></td></tr>
</table>

NB: ISNUMERIC DOES NOT EXIST ON HANA SO SOMETHING LIKE THIS NEED TO BE DONE IS YOU NEED IT: HTTP://STACKOVERFLOW.COM/QUESTIONS/27703567/WHAT-IS-THE-EQUIVALENT-OF-T-SQL-ISNUMERIC-FUNCTION-IN-HANA-SQLSCRIPT

Sample 3: The Problem

If there in the incorrect reference to non-existing database object in the code this can cause issue as well.



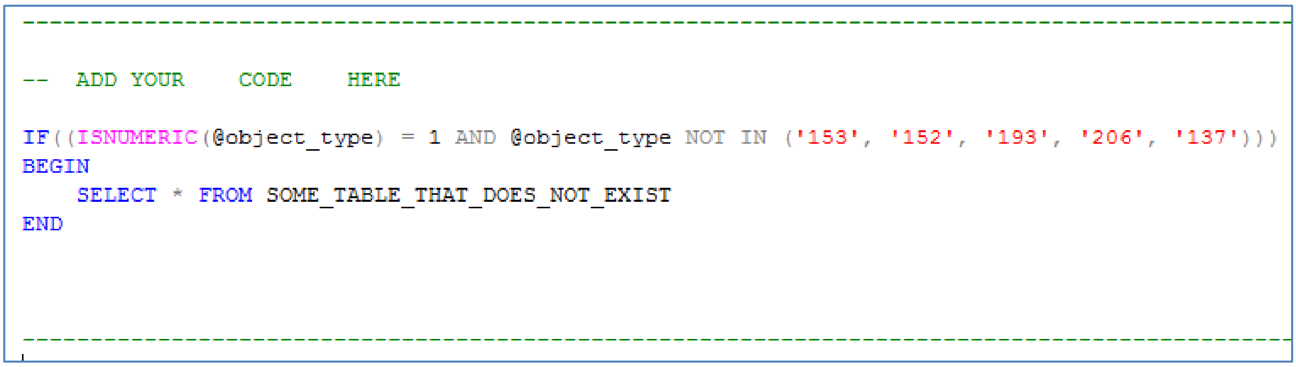
```
----------------------------------------------

--  ADD YOUR   CODE   HERE

SELECT * FROM SOME_TABLE_THAT_DOES_NOT_EXIST



----------------------------------------------
```

Here there are code that refer to a table that does not exist (Perhaps it existed at some point but was later removed but this modification referring to it was left there). In such a case no objects can be added at all in the SAP database and this code should be removed in 99% of the cases.

In very rare case this have been added as an add-on is about to make the tables but this modification can also affect the creation of the tables as the action of creating a table now also is sent through the SBO_SP_TransactionNotification and you end you with Problem 3 failing to create the user table/field/key/udo.

The Fix

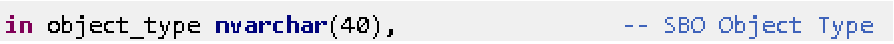
In such a case the ISNUMERIC(@object_Type) trick is not enough as UDT/UDF/UDK/UDO have object types '153', '152', '193' and '206' that are all numeric and also records are added to these tables (using @object_type '137'). For that reason you need to something like this if you can't remove the missing object reference.



```
--------------------------------------------------------------------------------------------

--  ADD YOUR   CODE   HERE

IF((ISNUMERIC(@object_type) = 1 AND @object_type NOT IN ('153', '152', '193', '206', '137')))
BEGIN
    SELECT * FROM SOME_TABLE_THAT_DOES_NOT_EXIST
END



--------------------------------------------------------------------------------------------
```

2. Incorrect object_type on HANA databases prior to SAP Business One 9.2 PL06 [HANA Only]

Please find the following line of code



```
in object_type nvarchar(40),                -- SBO Object Type
```

And make sure that the type is "nvarchar(40)"… if there instead of "nvarchar(40)" shows "nvarchar(20)" you have found the problem.

Correct	Incorrect

`in object_type nvarchar(40),`
`in transaction_type nchar(1),`
`in num_of_cols_in_key int,`
`in list_of_key_cols_tab_del varchar(255),`
`in list_of_cols_val_tab_del nvarchar(255)`
	`in object_type nvarchar(20),`
	`in transaction_type nchar(1),`
	`in num_of_cols_in_key int,`
	`in list_of_key_cols_tab_del varchar(255),`
	`in list_of_cols_val_tab_del nvarchar(255)`

To fix this issue create a new Create-script that have the right object_type length, Drop the old procedure and create the new corrected version.

NB: YOU CAN'T MODIFY A STORED PROCEDURE ON HANA, SO IN ORDER TO MODIFY IT YOU NEED TO DROP THE PROCEDURE AND RE-CREATE IT AGAIN WITH THE MODIFICATION. FOR THAT REASON IT IS NOT RECOMMENDED THAT IT IS DONE WHILE PEOPLE USE THE DATABASE!

3. Incorrect object_type on databases prior to SAP Business One 2005

Please find the following line of code

```
@object_type nvarchar(20),        -- SBO Object Type
```

And make sure that the type is “nvarchar(20)”… if there instead of “nvarchar(20)” shows “int” you have found the problem.

Correct	Incorrect

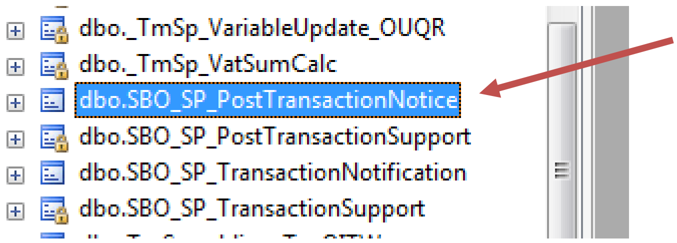
`@object_type nvarchar(20),`
`@transaction_type nchar(1),`
`@num_of_cols_in_key int,`
`@list_of_key_cols_tab_del nvarchar(255),`
`@list_of_cols_val_tab_del nvarchar(255)`
	`@object_type int,`
	`@transaction_type nchar(1),`
	`@num_of_cols_in_key int,`
	`@list_of_key_cols_tab_del nvarchar(255),`
	`@list_of_cols_val_tab_del nvarchar(255)`

To fix the problem simply replace the “int” with “nvarchar(20)” and

- Press F5 or the execute button to apply the change.

SBO_SP_TransactionNotification vs. SBO_SP_PostTransactionNotice

Prior to SAP Business One 8.8 the only place you were allowed to add SQL was the SBO_SP_TransactionNotification. In SBO 8.8 and higher SAP now also have a variant of the "SBO_SP_TransactionNotification" called "SBO_SP_PostTransactionNotice"



dbo._TmSp_VariableUpdate_OUQR
dbo._TmSp_VatSumCalc
dbo.SBO_SP_PostTransactionNotice
dbo.SBO_SP_PostTransactionSupport
dbo.SBO_SP_TransactionNotification
dbo.SBO_SP_TransactionSupport

New in 8.8

This stored procedure run after the normal "SBO_SP_TransactionNotification", but other than that they both have the same structure and can both hold the issue. If you have the issues mentioned in this document and that trying the above fixes you still have issue, you should check the "SBO_SP_PostTransactionNotice" for the same flaws using the same methods.

More information

If you like to know more, please check out the following SAP notes describing the problem on http://service.sap.com/notes:

- Note number: 967470
- Note number: 1070297
- Note number: 841475
- Note number: 2228417