

Serial Numbered Appliance Trading Organizations

This document is a technical reference containing all the queries and settings showing how WH9 was customized to fit a typical client and form a part of the micro-vertical solution for serial numbered appliance trader and similar companies.

1. User-Defined Fields

1.1. General Customization User Query

User query: bx_mobile_wh9_document_creation_type

```
SELECT CASE T0.[U_BXPDocTy] WHEN 20 THEN 1 ELSE 0 END as 'BXDOCTYP' FROM  
[dbo].[@BXPLMSMOBTHD] T0 WHERE T0.[Code] = [%3]
```

1.2. Goods Receipt

1.2.1. Goods Receipt PO Lines Screen

Add Customization Fields user table records:

FieldName	Visible	Screen
TextUoM	NO	GoodsReceiptPOLinesScreen

```
-- BXMobileWH9_GoodsReceiptPOLinesScreen_ButtonAdd_click  
-- Item code field must be scanned before Add button click  
IF $[TextItem] = ''  
SELECT 'You must scan an item first!' AS 'Message$', 'E' AS 'MessageType$',  
'TextItem' AS Click$
```

```
-- BXMobileWH9_GoodsReceiptPOLinesScreen_ButtonUpdate_click  
-- Item code field must be scanned before Update button click  
IF $[TextItem] = ''  
SELECT 'Please scan an item!' AS 'Message$', 'E' AS 'MessageType$',  
'TextItem' AS Click$
```

```
-- BXMobileWH9_GoodsReceiptPOLinesScreen_Activate  
-- Focus on Item field (when returning to this screen)  
SELECT 'TextItem' AS Click$
```

```
-- BXMobileWH9_GoodsReceiptPOLinesScreen_DataRepeater_InternalDataLoad  
-- Change contents of the list, show bar code instead of WH text  
SELECT
```



```

SUBSTRING(S.splitdata,1,50) AS [DataRepeater.UIItemCode],
C.CodeBars AS [DataRepeater.UIWarehouse]
FROM dbo.SplitStringForDataRepeater($[DataRepeater.UIItemCode]) S, OITM C
WHERE C.ItemCode=SUBSTRING(S.splitdata,1,PATINDEX('%*%',S.splitdata)-2)

```

```

-- BXMobilWH9_GoodsReceiptPOLinesScreen_TextItem_validate_after
-- After Item field has been entered write ' ' to quantity and press Add
button
-- (shortcut)
IF $[TextItem]<>' ' SELECT ' ' AS TextQuantity, 'ButtonAdd' AS Click$

```

1.2.2. Goods Receipt PO Quantities Serial Screen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	GoodsReceiptPOQuantitiesSerialScreen

```

-- BXMobilWH9_GoodsReceiptPOQuantitiesSerialScreen_Load
-- Load bin location for employee, fill EanKod with bar code
SELECT
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextBinLocation',
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2) ) AS
'EanKod'

```

```

--
BXMobilWH9_GoodsReceiptPOQuantitiesSerialScreen_TextBinLocation_validate_af
ter
-- Save the Bin Location value for next time
IF (SELECT ISNULL(OHEM.U_ToBL, ' ') FROM OHEM
WHERE empID = $[Employee.EmployeeID]) <> $[TextBinLocation]
UPDATE OHEM SET [U_ToBL]=$[TextBinLocation] WHERE empID =
$[Employee.EmployeeID]

```

```

--
BXMobilWH9_GoodsReceiptPOQuantitiesSerialScreen_TextSerial_validate_after
-- After serial number was specified press OK to go back
IF $[TextSerial]<>' ' SELECT 'OK' AS Click$

```

1.2.3. Goods Receipt PO Quantities Normal Screen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	GoodsReceiptPOQuantitiesNormalScreen
FieldName	Visible	Screen	

FieldName	Label	ReadOnly	Screen
TextUoM	NO	GoodsReceiptPOQuantitiesNormalScreen	

```
-- BXMmobileWH9_GoodsReceiptPOQuantitiesNormalScreen_Load
-- Load last location for employee, EAN bar code, set quantity to 1
-- and focus on quantity
SELECT
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2) ) AS
'EanKod',
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextBinLocation',
'1' AS 'TextQuantity' , 'TextQuantity' AS Click$
```

```
-- BXMmobileWH9_GoodsReceiptPOQuantitiesNormalScreen_Load
-- Load last location for employee, EAN bar code, set quantity to 1
-- and focus on quantity
SELECT
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2) ) AS
'EanKod',
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextBinLocation',
'1' AS 'TextQuantity' , 'TextQuantity' AS Click$
```

```
-- BXMmobileWH9_GoodsReceiptPOQuantitiesNormalScreen_ButtonAdd_click_after
-- After the Add button press OK to go back
SELECT 'OK' AS Click$
```

1.3. Free Goods Receipt

1.3.1. CreateGoodsReceiptPOLinesScreen

Add Customization Fields user table records:

FieldName	Visible	Screen
TextUoM	NO	GoodsReceiptReceiptLinesScreen

```
-- BXMmobileWH9_CreateGoodsReceiptPOLinesScreen_Load
-- Load saved warehouse for employee, focus on item
SELECT (SELECT OHEM.U_ToWh FROM OHEM WHERE empID = $[Employee.EmployeeID])
AS
'TextWarehouse', 'TextItem' AS Click$
```

```
-- BXMmobileWH9_CreateGoodsReceiptPOLinesScreen_Activate
-- Focus on item (after returning to this screen)
SELECT 'TextItem' AS Click$
```



```
-- BXMobilWH9_CreateGoodsReceiptPOLinesScreen_DataRepeater_InternalDataLoad
-- Modify list contents:
-- [DataRepeater.UIItem] Item code + name
-- [DataRepeater.UIWarehouse] Bar code (EAN)
SELECT SUBSTRING(S.splitdata,7,50) AS [DataRepeater.UIItem], C.CodeBars AS
[DataRepeater.UIWarehouse] FROM
dbo.SplitStringForDataRepeater($[DataRepeater.UIItem]) S,
OITM C WHERE
C.ItemCode=SUBSTRING(S.splitdata,7,PATINDEX('%*%',S.splitdata)-8)
```

```
-- BXMobilWH9_CreateGoodsReceiptPOLinesScreen_TextWarehouse_validate_after
-- Save warehouse for employee
IF (SELECT ISNULL(OHEM.U_ToWh, ''))
FROM OHEM WHERE empID = $[Employee.EmployeeID]) <> $[TextWarehouse]
UPDATE OHEM SET [U_ToWh]=$[TextWarehouse] WHERE empID =
$[Employee.EmployeeID]
```

```
-- BXMobilWH9_CreateGoodsReceiptPOLinesScreen_TextItem_validate
-- Check if item has been ordered / open quantity from this supplier
DECLARE @ckod nvarchar(MAX)
SET @ckod = $[TextItem]
IF NOT EXISTS (SELECT T0.ItemCode FROM OITM T0 WHERE T0.ItemCode=@ckod)
SELECT @ckod=T1.ItemCode FROM OBCD T1 WHERE T1.BcdCode=@ckod
IF @ckod<>' ' AND (SELECT ISNULL(SUM(T1.[OpenQty]),0) FROM OPOR T0 INNER JOIN
POR1 T1 ON
T0.[DocEntry] = T1.[DocEntry] WHERE T1.[ItemCode] =@ckod AND T0.[CardCode]
=LEFT($
[TextSupplier],PATINDEX('%*%', $[TextSupplier])-1))=0
SELECT 'There is no Purchase Order: ' + @ckod AS 'Message$', 'E' AS
'MessageType$'
ELSE SELECT ' ' AS dummy
```

```
-- BXMobilWH9_CreateGoodsReceiptPOLinesScreen_TextItem_validate
-- Check if item has been ordered / open quantity from this supplier
DECLARE @ckod nvarchar(MAX)
SET @ckod = $[TextItem]
IF NOT EXISTS (SELECT T0.ItemCode FROM OITM T0 WHERE T0.ItemCode=@ckod)
SELECT @ckod=T1.ItemCode FROM OBCD T1 WHERE T1.BcdCode=@ckod
IF @ckod<>' ' AND (SELECT ISNULL(SUM(T1.[OpenQty]),0) FROM OPOR T0 INNER JOIN
POR1 T1 ON
T0.[DocEntry] = T1.[DocEntry] WHERE T1.[ItemCode] =@ckod AND T0.[CardCode]
=LEFT($
[TextSupplier],PATINDEX('%*%', $[TextSupplier])-1))=0
SELECT 'There is no Purchase Order: ' + @ckod AS 'Message$', 'E' AS
'MessageType$'
ELSE SELECT ' ' AS dummy
```

```
-- BXMobilWH9_CreateGoodsReceiptPOLinesScreen_TextItem_validate_after
-- After item code has been entered write ' ' to quantity and press Add
IF $[TextItem]<>' ' SELECT ' ' AS TextQuantity, 'ButtonAdd' AS Click$
```


1.3.2. CreateGoodsReceiptPOQuantitiesSerialScreen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	CreateGoodsReceiptPOQuantitiesSerialScreen

```
-- BXMmobileWH9_CreateGoodsReceiptPOQuantitiesSerialScreen_Load
-- Load bin location from employee, fill EAN bar code
SELECT
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = ${Employee.EmployeeID}) AS
'TextBinLocation',
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING(${TextItem},1,PATINDEX('%*%',${TextItem})-2) ) AS
'EanKod'
```

```
--
BXMmobileWH9_CreateGoodsReceiptPOQuantitiesSerialScreen_TextBinLocation_validate_after
-- Save bin location to employee
IF (SELECT ISNULL(OHEM.U_ToBL, '') FROM OHEM WHERE empID =
${Employee.EmployeeID}) <> $
[TextBinLocation]
UPDATE OHEM SET [U_ToBL]=${TextBinLocation} WHERE empID =
${Employee.EmployeeID}
```

```
--
BXMmobileWH9_CreateGoodsReceiptPOQuantitiesSerialScreen_TextSerial_validate_after
-- After serial entry press OK to go back
IF ${TextSerial}<>' ' SELECT 'OK' AS Click$
```

1.3.3. CreateGoodsReceiptPOQuantitiesNormalScreen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	GoodsReceiptPOQuantitiesNormalScreen
FieldName	Visible	Screen	
TextUoM	NO	GoodsReceiptPOQuantitiesNormalScreen	

```
-- BXMmobileWH9_CreateGoodsReceiptPOQuantitiesNormalScreen_Load
-- Load last location for employee, EAN bar code, set quantity to 1
-- and focus on quantity
SELECT
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING(${TextItem},1,PATINDEX('%*%',${TextItem})-2) ) AS
'EanKod',
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = ${Employee.EmployeeID}) AS
```



```
'TextBinLocation',
'1' AS 'TextQuantity' , 'TextQuantity' AS Click$
```

```
--
BXMobleWH9_CreateGoodsReceiptPOQuantitiesNormalScreen_TextBinLocation_validate_after
-- Save bin location to employee
IF (SELECT ISNULL(OHEM.U_ToBL, '' ) FROM OHEM WHERE empID =
$[Employee.EmployeeID]) <> $
[TextBinLocation]
UPDATE OHEM SET [U_ToBL]=$[TextBinLocation] WHERE empID =
$[Employee.EmployeeID]
```

```
--
BXMobleWH9_CreateGoodsReceiptPOQuantitiesNormalScreen_ButtonAdd_click_after
-- After add button press OK to go back
SELECT 'OK' AS Click$
```

1.4. Transfer Stocks

1.4.1. TransferStocksScreen

Add Customization Fields user table records:

FieldName	Visible	Screen
TextBatch	NO	CreateGoodsReceiptPOQuantitiesSerialScreen

```
-- BXMobleWH9_TransferStocksScreen_DataRepeater_InternalDataLoad
-- Modify the list contents: show EAN bar code instead of item code
SELECT
SUBSTRING(S.splitdata,1,15)+'EAN:' + ISNULL(C.CodeBars, '')
AS [DataRepeater.UIItemCode]
FROM dbo.SplitStringForDataRepeater($[DataRepeater.UIItemCode]) S, OITM C
WHERE C.ItemCode=SUBSTRING(S.splitdata,1,PATINDEX('%*%',S.splitdata)-2)
```

```
-- BXMobleWH9_TransferStocksScreen_Activate
-- focus on item field after returning to this screen
SELECT 'TextItem' AS Click$
```

1.4.2. TransferStocksQuantitiesSerialScreen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	TransferStocksQuantitiesSerialScreen

```
-- BXMobleWH9_TransferStocksQuantitiesSerialScreen_Load
```



```
-- Fill to location from saved, load bar code into EAN code
SELECT
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = ${Employee.EmployeeID}) AS
'TextToLocation',
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING(${TextItem},1,PATINDEX('%*%',${TextItem})-2) ) AS
'EanKod'
```

```
--
BXMobileWH9_TransferStocksQuantitiesSerialScreen_TextToLocation_validate_after
-- Save to location for employee
IF (SELECT ISNULL(OHEM.U_ToBL, '') FROM OHEM WHERE empID =
${Employee.EmployeeID}) <> $
[TextToLocation]
UPDATE OHEM SET [U_ToBL]=${TextToLocation} WHERE empID =
${Employee.EmployeeID}
```

```
--
BXMobileWH9_TransferStocksQuantitiesSerialScreen_TextSerialNumber_validate_after
-- After seial number entered, press OK to go back
IF ${TextSerialNumber}<>' ' SELECT 'OK' AS Click$
```

```
--
BXMobileWH9_TransferStocksQuantitiesSerialScreen_TextSerialNumber_validate
-- Specify Mfr. Serial no if it has not been already done
DECLARE @sorsz nvarchar(MAX)
DECLARE @item nvarchar(MAX)
SET @sorsz = ${TextSerialNumber}
SET @item=SUBSTRING(${TextItem},1,PATINDEX('%*%',${TextItem})-2)
IF @sorsz<>' ' AND (SELECT FirmCode FROM OITM WHERE ItemCode=@item)=1 AND
(SELECT T0.MnfSerial FROM OSRN T0 WHERE T0.Distnumber=@sorsz AND
T0.ItemCode=@item)=''
IF ${GySz}=''
SELECT 'No Mfr. Serial no, continue without it?' AS 'Message$', 'YM' AS
'MessageType$'
ELSE
BEGIN
UPDATE OSRN SET MnfSerial = ${GySz} WHERE DistNumber=@sorsz AND
ItemCode=@item
SELECT '' AS 'GySz', 'Saved Mfr. Serial: '+ ${GySz} AS 'Message$', 'I' AS
'MessageType$'
END
```

```
-- BXMobileWH9_TransferStocksQuantitiesSerialScreen_GySz_validate
-- Focus on serial number after GySz field
SELECT 'TextSerialNumber' AS Click$
```


1.4.3. TransferStocksQuantitiesNormalScreen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	TransferStocksQuantitiesNormalScreen
FieldName	Visible	Screen	
TextUoM	NO	TransferStocksQuantitiesNormalScreen	

```
-- BXMobileWH9_TransferStocksQuantitiesNormalScreen_Load
-- load EAN bar codem to location, set quantity to 1 and focus on quantity
SELECT
(SELECT CodeBars FROM OITM WHERE
ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $
[TextItem])-2) ) AS 'EanKod',
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextToLocation',
'1' AS 'TextQuantity' , 'TextQuantity' AS Click$
```

```
--
BXMobileWH9_TransferStocksQuantitiesNormalScreen_TextToLocation_validate_after
-- Save to location for employee
IF (SELECT ISNULL(OHEM.U_ToBL, '') FROM OHEM WHERE empID =
$[Employee.EmployeeID]) <> $
[TextToLocation]
UPDATE OHEM SET [U_ToBL]=$[TextToLocation] WHERE empID =
$[Employee.EmployeeID]
```

```
-- BXMobileWH9_TransferStocksQuantitiesNormalScreen_ButtonAdd_click_after
-- Press OK to go back after Add button
SELECT 'OK' AS Click$
```

1.5. Mass Transfer

1.5.1. TransferStocksMassScreen

```
-- BXMobileWH9_TransferStocksMassScreen_Load
-- Load saved bin location from and to
SELECT
(SELECT OHEM.U_FromBL FROM OHEM
WHERE empID = $[Employee.EmployeeID]) AS 'TextWHBinLocationFrom',
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextWHBinLocationTo'
```

```
-- BXMobileWH9_TransferStocksMassScreen_TextWHBinLocationFrom_validate_after
-- Save from bin location
IF $[TextWHBinLocationFrom] <> '' AND (SELECT ISNULL(OHEM.U_FromBL, ''))
```



```
FROM OHEM WHERE empID = ${Employee.EmployeeID}) <> ${TextWHBinLocationFrom}
UPDATE OHEM SET [U_FromBL]=${TextWHBinLocationFrom} WHERE empID =
${Employee.EmployeeID}
```

```
-- BXMobilWH9_TransferStocksMassScreen_TextWHBinLocationFrom_validate_after
-- Save to bin location
IF ${TextWHBinLocationTo} <> '' AND (SELECT ISNULL(OHEM.U_ToBL, '') FROM OHEM
WHERE empID =
${Employee.EmployeeID}) <> ${TextWHBinLocationTo}
UPDATE OHEM SET [U_ToBL]=${TextWHBinLocationTo} WHERE empID =
${Employee.EmployeeID}
```

```
-- BXMobilWH9_TransferStocksMassScreen_DataRepeater_InternalDataLoad
-- Modify list contents
-- [DataRepeater.UIItemCode] Item code + name
-- [DataRepeater.UIOnHandQuantity] Bar code (EAN) + quantity
SELECT C.CodeBars + ' ' +SUBSTRING(S.splitdata2,1,50) AS
[DataRepeater.UIOnHandQuantity]
FROM dbo.SplitStringForDataRepeater2(${DataRepeater.UIItemCode}, $
[DataRepeater.UIOnHandQuantity]) S, OITM C
WHERE C.ItemCode=SUBSTRING(S.splitdata,1,PATINDEX('%*%',S.splitdata)-1)
```

1.6. Transfer Stock Request

1.6.1. TransferStockRequestLineScreen

Add Customization Fields user table records:

FieldName	Visible	Screen
TextBatch	NO	TransferStockRequestLineScreen

```
-- BXMobilWH9_TransferStockRequestLineScreen_DataRepeater_InternalDataLoad
-- Modify list contents, add EAN bar code
SELECT SUBSTRING(S.splitdata,7,15)+' EAN:'+ ISNULL(C.CodeBars, '') AS
[DataRepeater.UIItem]
FROM dbo.SplitStringForDataRepeater(${DataRepeater.UIItem}) S, OITM C
WHERE C.ItemCode=SUBSTRING(S.splitdata,7,PATINDEX('%*%',S.splitdata)-8)
```

```
-- BXMobilWH9_TransferStockRequestLineScreen_TextItem_validate_after
-- After Item Code has been entered, set quantity to '' and press Add
IF ${TextItem}<>'' SELECT '' AS TextQuantity, 'ButtonAdd' AS Click$
```

```
-- BXMobilWH9_TransferStockRequestLineScreen_ButtonAdd_click
-- Item code scanning is mandatory (before pressing Add button)
IF ${TextItem} =''
SELECT 'You must scan an item!' AS 'Message$', 'E' AS 'MessageType$',
'TextItem'
AS Click$
```



```
-- BXMobileWH9_TransferStockRequestLineScreen_ButtonUpdate_click
-- Item code scanning is mandatory (before pressing Update button)
IF $[TextItem] = ''
SELECT 'You must scan an item!' AS 'Message$', 'E' AS 'MessageType$',
'TextItem'
AS Click$
```

```
-- Visszatéréskor fókusz a cikkszámra
-- BXMobileWH9_TransferStockRequestLineScreen_Activate
SELECT 'TextItem' AS Click$
```

1.6.2. TransferStockRequestsQuantitiesSerialScreen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	TransferStockRequestsQuantitiesSerialScreen

```
-- BXMobileWH9_TransferStockRequestsQuantitiesSerialScreen_Load
-- Set to location from saved, set EAN code from barcode
SELECT
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextToLocation',
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2) ) AS
'EanKod'
```

```
--
BXMobileWH9_TransferStockRequestsQuantitiesSerialScreen_TextToLocation_validate_after
-- Save to location if changed
IF (SELECT ISNULL(OHEM.U_ToBL, '') FROM OHEM WHERE empID =
$[Employee.EmployeeID])
<> $[TextToLocation]
UPDATE OHEM SET [U_ToBL]=$[TextToLocation] WHERE empID =
$[Employee.EmployeeID]
```

```
--
BXMobileWH9_TransferStockRequestsQuantitiesSerialScreen_TextSerialNumber_validate_after
-- After serial number has been entered press OK to go back
IF $[TextSerialNumber]<>' ' SELECT 'OK' AS Click$
```

```
--
BXMobileWH9_TransferStockRequestsQuantitiesSerialScreen_TextSerialNumber_validate
-- Must specify manufacturer's serial number if it wasn't already done
DECLARE @sorsz nvarchar(MAX)
DECLARE @item nvarchar(MAX)
```



```

SET @sorsz = $[TextSerialNumber]
SET @item=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2)
IF @sorsz<>' ' AND (SELECT FirmCode FROM OITM WHERE ItemCode=@item)=1 AND
(SELECT T0.MnfSerial FROM OSRN T0 WHERE T0.Distnumber=@sorsz AND
T0.ItemCode=@item)=''
IF $[GySz]=' '
SELECT 'No Mfr. Serial, continue without?' AS 'Message$', 'YM' AS
'MessageType$'
ELSE
BEGIN
UPDATE OSRN SET MnfSerial = $[GySz] WHERE DistNumber=@sorsz AND
ItemCode=@item
SELECT '' AS 'GySz', 'Recorded Mfr.Serial: ' + $[GySz] AS 'Message$', 'I' AS
'MessageType$'
END

```

```

--
BXMobileWH9_TransferStockRequestsQuantitiesSerialScreen_GySz_validate_after
-- Focus on Serial no after GySz field
SELECT 'TextSerialNumber' AS Click$

```

1.6.3. TransferStockRequestsQuantitiesNormalScreen

Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	TransferStockRequestsQuantitiesNormalScreen
FieldName	Visible	Screen	
TextUoM	NO	TransferStockRequestsQuantitiesNormalScreen	

```

-- BXMobileWH9_TransferStockRequestsQuantitiesNormalScreen_Load
-- fill EanKod, To location, set 1 to quantity and focus on quantity
SELECT
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2) ) AS
'EanKod',
(SELECT OHEM.U_ToBL FROM OHEM WHERE empID = $[Employee.EmployeeID]) AS
'TextToLocation',
'1' AS 'TextQuantity' , 'TextQuantity' AS Click$

```

```

--
BXMobileWH9_TransferStockRequestsQuantitiesNormalScreen_TextToLocation_valid
ate_after
-- Save to bin location for next time
IF (SELECT ISNULL(OHEM.U_ToBL, ' ') FROM OHEM WHERE empID =
$[Employee.EmployeeID])
<> $
[TextToLocation]
UPDATE OHEM SET [U_ToBL]=$[TextToLocation] WHERE empID =

```



```
$[Employee.EmployeeID]
```

```
--
BXMobileWH9_TransferStockRequestsQuantitiesNormalScreen_ButtonAdd_click_after
-- After Add press OK to go back
SELECT 'OK' AS Click$
```

1.7. Picking Screen

1.7.1. Picking Screen

Show TourID field from PickList head and this will be written to the Delivery document. Also add new field: Hely (place). Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
BO_U_Tour ID	TuraID	YES	PickingLinesScreen
FieldName	Label	Screen	
Hely	Place	PickingLinesScreen	

```
-- BXMobileWH9_PickingScreen_TextPickListNo_validate_after
-- Trick to make sure pick list bin locations are changed if needed
IF $[TextPickListNo]<>' ' AND $[Hely]<>' '
UPDATE PKL2 SET BinAbs=(SELECT T0.AbsEntry
FROM OBIN T0 WHERE BinCode=$[Hely]) WHERE AbsEntry=CAST($[TextPickListNo] AS
INT)
```

```
-- BXMobileWH9_PickingScreen_Hely_validate_after
-- Trick to make sure pick list bin locations are changed if needed
IF $[TextPickListNo]<>' ' AND $[Hely]<>' '
UPDATE PKL2 SET BinAbs=(SELECT T0.AbsEntry FROM OBIN T0 WHERE
BinCode=$[Hely])
WHERE AbsEntry=CAST($[TextPickListNo] AS INT)
```

```
-- BXMobileWH9_PickingScreen_DataRepeater_InternalDataLoad
-- Show TourID in the list in the place of date field
SELECT SUBSTRING(S.splitdata2,1,50)+ ' Tour: ' + C.U_TourID AS
[DataRepeater.UIPickDate]
FROM dbo.SplitStringForDataRepeater2($[DataRepeater.UIPickListNo], $
[DataRepeater.UIPickDate]) S, OPKL C
WHERE C.AbsEntry= CAST(SUBSTRING(S.splitdata, 11,5) AS INT)
```

1.7.2. Picking Lines Screen

Show TourID field, change bin code to not read only. Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
BO_U_TourID	TuraID	YES	PickingLinesScreen
TextBinCode	Item	NO	PickingLinesScreen

```
-- BXMobileWH9_PickingLinesScreen_Load
-- Fill TourID field on top of the screen
SELECT
(SELECT OPKL.U_TourID FROM OPKL WHERE OPKL.AbsEntry = CAST(LEFT($
[TextPickList],PATINDEX('%*%', $[TextPickList])-1) AS INT)) AS [BO_U_TourID]
```

```
-- BXMobileWH9_PickingLinesScreen_OK_clicked
-- Mandatory to scan/enter item in bin code field
IF $[TextBinCode] = ''
SELECT 'Scan an item!' AS 'Message$', 'E' AS 'MessageType$', 'TextBinCode'
AS
Click$
```

```
-- BXMobileWH9_PickingLinesScreen_DataRepeater_InternalDataLoad
-- Show EAN bar code in list
SELECT ' EAN: ' + C.CodeBars AS [DataRepeater.UIBinCode]
FROM dbo.SplitStringForDataRepeater2($[DataRepeater.UIItem], $
[DataRepeater.UIBaseType]) S, OITM C
WHERE C.ItemCode=SUBSTRING(S.splitdata,1,PATINDEX('%*%', S.splitdata)-1)
```

```
-- BXMobileWH9_PickingLinesScreen_Activate
-- Focus on bin code field
SELECT 'TextBinCode' AS Click$
```

1.7.3. PickingLinesPickSerialScreen

Add new EanKod field to screen, set bin field to read only. Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	PickingLinesPickSerialScreenw
FieldName	Label	Screen	
TextBinLocation	YES	PickingLinesPickSerialScreen	

```
-- BXMobileWH9_PickingLinesPickSerialScreen_Load
SELECT $[TextRecBin] AS 'TextBinLocation',
(SELECT CodeBars FROM OITM
WHERE ItemCode=SUBSTRING($[TextItem],1,PATINDEX('%*%', $[TextItem])-2) ) AS
'EanKod'
```

```
-- Sorozatszám megadása után <Könyvelés> gomb megnyomása:
-- BXMobileWH9_PickingLinesPickSerialScreen_TextSerialNumber_validate_after
IF $[TextSerialNumber]<>' '
IF CAST(LEFT($[TextOpenQty],PATINDEX('% %', $[TextOpenQty]+' ')-1) AS INT)=0
SELECT 'OK' AS 'Click$'
```



```
ELSE SELECT 'OK##Option' AS 'Click$'
```

1.7.4. PickingLinesPickNormalScreen

Hide UoM field from screen. Add Customization Fields user table records:

FieldName	Label	ReadOnly	Screen
EanKod	EAN	YES	PickingLinesPickNormalScreen
FieldName	Visible	Screen	
TextUoM	NO	PickingLinesPickNormalScreen	
FieldName	ReadOnly	Screen	
TextBinLocation	YES	PickingLinesPickNormalScreen	

```
-- BXMobilWH9_PickingLinesPickNormalScreen_Load
SELECT ${TextRecBin} AS 'TextBinLocation',
(SELECT CodeBars FROM OITM
WHERE ItemCode=${SelectedPickListLine.ItemCode}) AS 'EanKod',1 AS
'TextQuantity',
'TextQuantity' AS Click$
```

```
-- BXMobilWH9_PickingLinesPickNormalScreen_TextQuantity_validate_after
-- After quantity has been entered, press OK to book and then back button
IF CAST(LEFT(${TextOpenQty},PATINDEX('% %',${TextOpenQty}+' ')-1) AS
INT)-CAST($
[TextQuantity] AS INT)<=0 SELECT 'OK' AS 'Click$'
ELSE SELECT 'OK##Option' AS 'Click$'
```

1.8. Goods Receipt

1.8.1. GoodsReceiptLinesScreen

Hide UoM field from screen (Add Customization Fields user table records)

FieldName	Visible	Screen
TextUoM	NO	PickingLinesScreen

```
-- bx_mobile_wh9_goodsreceipt_pricing
-- Sets price, currency and account no fields,
-- see WH9 documentation for reference
SELECT ISNULL(T1.Price,0) AS 'BXITPRC', ISNULL(T1.Currency,'FT') AS
'BXITCURR',
'5911' AS 'BXITACCN' FROM OITM T0
JOIN ITM1 T1 ON T0.ItemCode=T1.ItemCode AND T1.Pricelist='1'
WHERE T0.[ItemCode] =[%3]
```

```
-- BXMobilWH9_GoodsReceiptLinesScreen_Load
-- Load warehouse from employee's saved value, focus on item
```



```
SELECT
(SELECT OHEM.U_ToWh FROM OHEM WHERE empID = ${Employee.EmployeeID}) AS
'TextWarehouse',
'TextItem' AS Click$
```

```
-- BXMmobileWH9_GoodsReceiptLinesScreen_TextWarehouse_validate_after
-- Save warehouse code for next time
IF (SELECT ISNULL(OHEM.U_ToWh, '') FROM OHEM
WHERE empID = ${Employee.EmployeeID}) <> ${TextWarehouse}
UPDATE OHEM SET [U_ToWh]=${TextWarehouse} WHERE empID =
${Employee.EmployeeID}
```

```
-- BXMmobileWH9_GoodsReceiptLinesScreen_TextItem_validate_after
-- After item code has been entered, automatically
-- set quantity to ' ' and press add
IF ${TextItem}<>' ' SELECT
' ' AS TextQuantity,
'ButtonAdd' AS Click$
```

```
-- BXMmobileWH9_GoodsReceiptLinesScreen_DataRepeater_InternalDataLoad
-- Change the contents of the list
-- [DataRepeater.UIItem] Item code + name
-- [DataRepeater.UIWarehouse] Bar code (EAN)
SELECT SUBSTRING(S.splitdata,7,50) AS [DataRepeater.UIItem],
C.CodeBars AS [DataRepeater.UIWarehouse]
FROM dbo.SplitStringForDataRepeater(${DataRepeater.UIItem}) S,
OITM C WHERE
C.ItemCode=SUBSTRING(S.splitdata,7,PATINDEX('%*%',S.splitdata)-8)
```

```
-- BXMmobileWH9_GoodsReceiptLinesScreen_Activate
-- Focus on item code (on return to this screen)
SELECT 'TextItem' AS Click$
```

1.9. Query Stocks

1.9.1. QueryStocksScreen

Hide batch field from screen (Add Customization Fields user table records)

FieldName	Visible	Screen
TextBatch	NO	QueryStocksScreen

```
-- BXMmobileWH9_QueryStocksScreen_Load
-- Load default warehouse for employee, focus on Item field
SELECT (SELECT OHEM.U_ToWh FROM OHEM WHERE empID = ${Employee.EmployeeID})
AS
'TextWHBinLocation', 'TextItem' AS Click$
```



```
-- BXMmobileWH9_QueryStocksScreen_DataRepeater_InternalDataLoad
-- Modify list contents (show bar code)
SELECT SUBSTRING(S.splitdata,1,50) +' VK: ' + C.CodeBars AS
[DataRepeater.UIItemCode]
FROM dbo.SplitStringForDataRepeater($[DataRepeater.UIItemCode]) S, OITM C
WHERE C.ItemCode=SUBSTRING(S.splitdata,1,PATINDEX('%*%',S.splitdata)-1)
```

```
-- BXMmobileWH9_QueryStocksScreen_TextItem_validate_after
-- Save bar code (new) for item into OITM.U_EanKod if the item code is also
filled
DECLARE @ckod nvarchar(MAX)
SET @ckod = $[TextItem]
IF NOT EXISTS (SELECT T0.ItemCode FROM OITM T0 WHERE T0.ItemCode=@ckod)
SELECT @ckod=T1.ItemCode FROM OBCD T1 WHERE T1.BcdCode=@ckod
IF @ckod<>' ' AND ISNULL((SELECT OITM.CodeBars FROM OITM WHERE
OITM.ItemCode=@ckod),'')<>' '
SELECT OITM.CodeBars AS 'EanKod' FROM OITM WHERE OITM.ItemCode=@ckod
IF @ckod<>' ' AND ISNULL((SELECT OITM.CodeBars FROM OITM WHERE
OITM.ItemCode=@ckod),'')=''
SELECT OITM.U_EanKod AS 'EanKod' FROM OITM WHERE OITM.ItemCode=@ckod
```

```
-- BXMmobileWH9_QueryStocksScreen_EanKod_validate
IF $[TextItem]<>' ' AND $[EanKod]=' '
SELECT 'Please specify EAN code!' AS 'Message$', 'E' AS 'MessageType$'
```

```
-- BXMmobileWH9_QueryStocksScreen_EanKod_validate_after
-- save new EAN code to OITM.U_EanKod
DECLARE @ckod nvarchar(MAX)
SET @ckod = $[TextItem]
IF NOT EXISTS (SELECT T0.ItemCode FROM OITM T0 WHERE T0.ItemCode=@ckod)
SELECT @ckod=T1.ItemCode FROM OBCD T1 WHERE T1.BcdCode=@ckod
IF $[EanKod]<>' ' AND @ckod<>' ' AND
ISNULL((SELECT OITM.CodeBars FROM OITM WHERE OITM.ItemCode=@ckod),'')=''
UPDATE OITM SET [U_EanKod] = $[EanKod] WHERE OITM.[ItemCode] = @ckod
IF $[EanKod]<>' ' AND ISNULL((SELECT OITM.CodeBars FROM OITM WHERE
OITM.ItemCode=@ckod),'')=''
SELECT 'EAN kód saved: '+ $[EanKod] AS 'Message$', 'I' AS 'MessageType$'
```

1.9. Packing

1.9.1. PackingScreen (Print labels)

Show a Label field on screen (Add Customization Fields user table records)

FieldName	Labels	Screen
RptCode	Labels	PackingScreen

```
-- BXMmobileWH9_PackingScreen_RPTCode_validate
```



```
-- If Label field is not empty then print (print layout for delivery document labels)
IF ${RPTCode}<>' ' AND ${TextDeliveryDocNum}<>' '
SELECT 'Print labels: ' + ${RPTCode} AS 'Message$', 'YM' AS 'MessageType'
```

```
-- BXMobileWH9_PackingScreen_RPTCode_validate_after
-- Print using Crystal Reports defined example to PrimoPDF printer with DocKey@ parameter
IF ${RPTCode}<>' ' AND ${TextDeliveryDocNum}<>' '
SELECT ${RPTCode} AS PrintLayout$,
(SELECT DocEntry FROM ODLN WHERE DocNum=${TextDeliveryDocNum}) AS
[Print_DocKey@],
'PrimoPDF' AS PrintPrinter$,
'Report print: ' + ${RPTCode} AS 'Message$', 'I' AS 'MessageType'
```

2. Appendix

2.1. User fields in SAP

By default Start / Target warehouses and storage the Employees (OHEM) table:

- OHEM.U_FromBL
- OHEM.U_ToBL
- OHEM.U_ToWh
- OHEM-U_FromWh

The master item data EAN temporary location:

OITM.U_EanKod EanKod Alfanumerikus(16)

U_TourID sets the storage application for Pick List header and the delivery notes header.

2.2. All Customization Fields

CustomizationFields													
#	Code	Name	Data Repeater	Field Display Type	Field Name	Field Type	Label	Language Code	Module	On External Form	Read Only	Screen	Visible
1	10	uf8	None	None	BO_U_TourID	String	TourID		EXMobileWH9	No	Yes	PickingLinesScreen	Yes
2	12	uf12	None	None	BO_NumAtCard	String	Ref.No.		EXMobileWH9	No	No	CreateGoodsReceiptPOScreen	Yes
3	13	uf13	None	None	TextBinCode	String	Item		EXMobileWH9	No	No	PickingLinesScreen	Yes
4	14	uf14	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	GoodsReceiptLinesScreen	No
5	15	uf15	None	None	TextBatch	String	No Visible		EXMobileWH9	No	No	QueryStocksScreen	No
6	16	uf16	None	None	EanKod	String	EAN		EXMobileWH9	No	No	QueryStocksScreen	Yes
7	17	uf17	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	PickingLinesPickSerialScreen	Yes
8	18	uf18	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	GoodsReceiptLinesScreen	No
9	19	uf19	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	GoodsReceiptPOQuantitiesSerialScreen	Yes
10	20	uf20	None	None	TextBatch	String	No Visible		EXMobileWH9	No	No	TransferStocksScreen	No
11	21	uf21	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	TransferStocksQuantitiesSerialScreen	Yes
12	22	uf22	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	TransferStocksRequestLineScreen	No
13	23	uf23	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	SalesReturnLinesScreen	No
14	24	uf24	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	SalesReturnQuantitiesSerialScreen	Yes
15	25	uf25	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	CreateGoodsReceiptPOQuantitiesNormalScreen	No
16	26	uf26	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	CreateGoodsReceiptPOQuantitiesNormalScreen	Yes
17	27	uf27	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	GoodsReceiptPOQuantitiesNormalScreen	No
18	28	uf28	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	GoodsReceiptPOQuantitiesNormalScreen	Yes
19	29	uf29	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	TransferStocksQuantitiesNormalScreen	Yes
20	30	uf30	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	TransferStocksQuantitiesNormalScreen	No
21	31	uf31	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	PickingLinesPickNormalScreen	Yes
22	32	uf32	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	PickingLinesPickNormalScreen	No
23	33	uf33	None	None	TextBinLocation	String	BinLoc		EXMobileWH9	No	Yes	PickingLinesPickNormalScreen	Yes
24	34	uf34	None	None	TextBinLocation	String	BinLoc		EXMobileWH9	No	Yes	PickingLinesPickSerialScreen	Yes
25	35	uf35	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	TransferStockRequestsQuantitiesSerialScreen	Yes
26	36	uf36	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	TransferStockRequestsQuantitiesNormalScreen	Yes
27	37	uf37	None	None	TextUoM	String	No Visible		EXMobileWH9	No	No	TransferStockRequestsQuantitiesNormalScreen	No
28	38	uf38	None	None	Hely	String	BinLoc		EXMobileWH9	No	No	PickingScreen	Yes
29	39	uf39	None	None	GySz	String	Mfr. No.		EXMobileWH9	No	No	TransferStocksQuantitiesSerialScreen	Yes
30	4	uf4	None	None	EanKod	String	EAN		EXMobileWH9	No	Yes	CreateGoodsReceiptPOQuantitiesSerialScreen	Yes
31	40	uf40	None	None	BO_DocDueDate	String	Delivery		yyy	No	No	PickingLinesScreen	Yes
32	5	uf1	None	None	TextUoM	String	No Visible		EXMobileWH9	No	Yes	CreateGoodsReceiptPOLinesScreen	No
33	8	uf6	None	None	RPTCode	String	Labels		EXMobileWH9	No	No	PackingScreen	Yes
34			None	None		String				No	No		Yes

2.3. Helper SQL procedures

```
-- This function is a helper for WH9 list splitting
CREATE FUNCTION [dbo].[SplitStringForDataRepeater]
( @string NVARCHAR(MAX) )
RETURNS @output TABLE(splitdata NVARCHAR(MAX) )
BEGIN
DECLARE @START INT, @END INT
SELECT @START = 1, @END = CHARINDEX('##', @string)
WHILE @START < LEN(@string) + 1 BEGIN
IF @END = 0
SET @END = LEN(@string) + 2
INSERT INTO @output (splitdata)
VALUES(SUBSTRING(@string, @START, @END - @START))
SET @START = @END + 2
SET @END = CHARINDEX('##', @string, @START)
END
RETURN
END
```

```
-- This function is a helper for WH9 list splitting
CREATE FUNCTION [dbo].[SplitStringForDataRepeater2]
( @string NVARCHAR(MAX), @string2 NVARCHAR(MAX) )
RETURNS @output TABLE(splitdata NVARCHAR(MAX) , splitdata2 NVARCHAR(MAX))
BEGIN
DECLARE @START INT, @END INT , @start2 INT, @end2 INT
SELECT @START = 1, @END = CHARINDEX('##', @string) , @start2 = 1, @end2 =
CHARINDEX('##', @string2)
WHILE @START < LEN(@string) + 1 BEGIN
IF @END = 0
BEGIN
```



```
SET @END = LEN(@string) + 2
SET @end2 = LEN(@string2) + 2
END
INSERT INTO @output (splitdata, splitdata2)
VALUES(SUBSTRING(@string, @START, @END - @START),SUBSTRING(@string2,
@start2, @end2 - @start2))
SET @START = @END + 2
SET @END = CHARINDEX('##', @string, @START)
SET @start2 = @end2 + 2
SET @end2 = CHARINDEX('##', @string2, @start2)
END
RETURN
END
```

From:

<http://wiki.produmex.name/> -

Permanent link:

<http://wiki.produmex.name/doku.php?id=implementation:bxmobwh:fullref>



Last update: **2017/05/19 10:22**