

Notes on LHA Implementation of DAXKitting

Note

Not all the changes in this project are included. It just shows what I did to implement it in CU13 with the Retail module enabled. There were some objects and methods that needed removed from the var layer and re-built.

Project: _KIT

Deployed to R3 DEV

See <https://docs.microsoft.com/en-us/dynamicsax-2012/appuser-itpro/how-to-export-and-import-a-model>

- Used this AX Management Shell command: install-axmodel -file C:\temp\Kit_45_ax2012R3CU9.axmodel -Conflict Push -Config LHA_2012R3_DEV
- The model file is in releases/0
- Axmodel file has to be unblocked before the install-axmodel command can run. See <http://yetanotherdynamicsaxblog.blogspot.com/2014/04/error-when-importing-model-downloaded.html>

```
PS C:\temp> Install-AXModel -File C:\temp\Kit_45_ax2012R3CU9.axmodel -Conflict Push -Config LHA_2012R3_DEV

The model is not signed. Are you sure you want to install this model (Y/N)?
Continue with this operation?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
The model file C:\temp\Kit_45_ax2012R3CU9.axmodel was successfully imported into the Var layer.
Imported model: KIT (CodePlex).
\Classes: 43
\Data Dictionary\Base Enums: 3
\Data Dictionary\Extended Data Types: 11
\Data Dictionary\Tables: 12
\Forms: 12
\Jobs: 1
\Label Files: 1
\Macros: 1
\Menu items\Display: 3
\Menus: 2
\Projects\Shared: 3
\Security\Duties: 1
\Security\Privileges: 1
\Security\Process Cycles: 1
\Security\Roles: 1

Because of the changes made to the model store, it is highly recommended to restart the AOS, run appropriate scripts, and compile and synchronize the application. Not doing that may cause runtime errors and loss of data.
PS C:\temp>
```

After installing module:

- Restarted AOS 4 (on db3-01)
- Opened DEV client
- Selected “compile and synchronize”



- Shoot... should have hit skip and used AXBuild to compile....
- Killed AX client
- Opened elevated command prompt
- Went to C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_DEV\bin>
- Ran: axbuild xppcompileall /s=04

```
C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_DEV\bin>axbuild xppc
ompileall /s=04

Microsoft (R) Dynamics AX Build Utility 6.3.164.0
Copyright (C) Microsoft Corporation. All rights reserved.

*** Compile all starting at 5/28/2020 9:00:16 AM.
*** Kicking off header compilation (pass 1) with 24 workers.
*** Kicking off full compilation (pass 2) with 24 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Kicking off full compilation (pass 3) with 24 workers.
*** Kicking off full compilation (pass 4) with 1 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Compile all completed at 5/28/2020 9:21:04 AM. Total duration was 00:20:47.6
659400.

Log files are located at 'C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2
012R3_DEV\Log'

C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_DEV\bin>
```

- Opened AX client and ran full db sync
- Model is imported into the var layer, “Kitting” model

About the Code

- The project _KIT appears to hold the objects that were modified / added.
- Added the duty “Kit” to the Everyone group

Modifications

- Added Kits duty to Everyone role
- Added reference to Kitting menu in Sales & Marketing

Deployment to GOLD

PS C:\temp> Install-AXModel -File C:\temp\Kit_45_ax2012R3CU9.axmodel -Conflict Push -Config LHA_2012R3_GOLD

```
PS C:\temp> Install-AXModel -File C:\temp\Kit_45_ax2012R3CU9.axmodel -Conflict Push -Config LHA_2012R3_GOLD

The model is not signed. Are you sure you want to install this model (Y/N)?
Continue with this operation?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
The model file C:\temp\Kit_45_ax2012R3CU9.axmodel was successfully imported into the Var layer.
Imported model: KIT (CodePlex).
\Classes: 43
\Data Dictionary\Base Enums: 3
\Data Dictionary\Extended Data Types: 11
\Data Dictionary\Tables: 12
\Forms: 12
\Jobs: 1
\Label Files: 1
\Macros: 1
\Menu items\Display: 3
\Menus: 2
\Projects\Shared: 3
\Security\Duties: 1
\Security\Privileges: 1
\Security\Process Cycles: 1
\Security\Roles: 1

Because of the changes made to the model store, it is highly recommended to restart the AOS, run appropriate scripts, and compile and synchronize the application. Not doing that may cause runtime errors and loss of data.
PS C:\temp>
```

Restart-service 'AOS60-\$03'

C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>axbuild xpcompileall /s=03

```
C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>axbuild xpcompileall /s=03

Microsoft (R) Dynamics AX Build Utility 6.3.164.0
Copyright (C) Microsoft Corporation. All rights reserved.

*** Compile all starting at 5/28/2020 10:36:10 AM.
*** Kicking off header compilation (pass 1) with 24 workers.
*** Kicking off full compilation (pass 2) with 24 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Kicking off full compilation (pass 3) with 24 workers.
*** Kicking off full compilation (pass 4) with 1 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Compile all completed at 5/28/2020 10:52:35 AM. Total duration was 00:16:25.4395341.

Log files are located at 'C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\Log'

C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>
```

Opened Gold client, selected Skip, ran full db sync.

Ran build script

```
Model synchronized.  
Compiling AX X++ ...  
AX X++ compiled.  
Checking for compile errors ...  
Compile errors found  
Press any key to continue . . .
```

Compiler Error log : 5/28/2020

Errors : 16

Warnings : 16

Best Practice Deviations : 0

Severity	Path
Error	\Classes\SalesFormletterParmDataPackingslip\createParmSublineFromWHSPParameters
Error	\Classes\SalesFormletterParmDataPackingslip\initRecordLists
Error	\Classes\SalesFormletterParmDataPackingslip\insertParmLine
Error	\Classes\SalesFormletterParmDataPackingslip\insertRecordLists
Error	\Classes\SalesFormletterParmDataPackingslip\intercompanySalesAtCostPriceHandler
Error	\Forms\PurchTable\Designs\DesignList\PISCOFINSFixedAssetCreditAppropMethod_BR
Error	\Forms\PurchTable\Designs\DesignList\PISCOFINSFixedAssetCreditGroup_BR
Error	\Forms\PurchTable\Designs\DesignList\PISCOFINSFixedAssetCreditInstallments_BR
Error	\Forms\PurchTable\Designs\DesignList\PISCOFINSFixedAssetCreditPurpose_BR
Error	\Forms\SalesQuotationProjTable\Methods\isCalcTax
Error	\Forms\SalesQuotationTable\Designs\DesignList\KitShowLinesBtn\Methods\clicked
Error	\Forms\SalesTable\Data Sources\SalesTable\Methods\create
Error	\Forms\SalesTable\Data Sources\SalesTable\Methods\executeQuery
Error	\Forms\SalesTable\Data Sources\SalesTable\Methods\leave
Error	\Forms\SalesTable\Data Sources\SalesTable\Methods\research
Error	\Forms\SalesTable\Methods\task
Warning	\Data

excludeContainerFilters

Find: Methods

Name & location

Date

Advanced

Filter

Search:

Methods

Named:

Containing text:

excludeContainerFilters

Look in:

\Classes\SalesFormletterParmDataPackingslip

Name	From the source	Path
createParmSublineFromWHSPParameters	[23] if (!excludeContainerFilters)	\Classes\SalesFormletterParmDataP...
createParmSublineFromWHSPParameters	[25] excludeContainerFilters = new Set(Types::C...	
createParmSublineFromWHSPParameters	[28] if (!excludeContainerFilters.in([_loadLineRecId,...	
createParmSublineFromWHSPParameters	[31] excludeContainerFilters.add([_loadLineRecId...	

Class extends SalesFormletterParmData

Variable is not in that class.

Searched FormletterParmData... not in there either.

Checked method in R3 test environment.

There it is...

\Classes\SalesFormLetterParmDataPackingslip - Editor

classDeclaration
createParmSublineFromWHSPParameters

```
class SalesFormLetterParmDataPackingslip
extends SalesFormLetterParmData
implements FormLetterParmDataCorrectable
{
    boolean setProforma;
    boolean proformaIsSet;

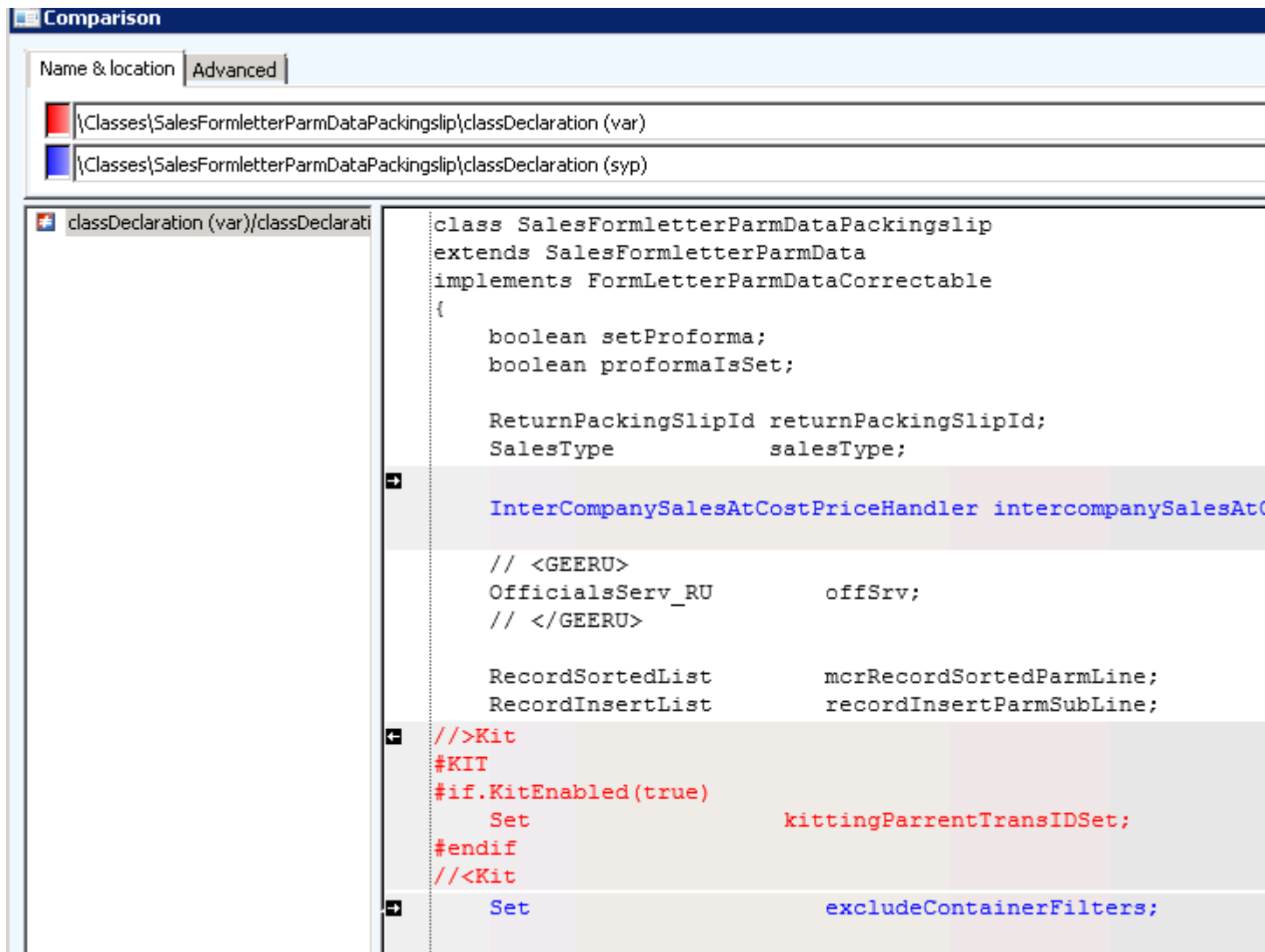
    ReturnPackingSlipId returnPackingSlipId;
    SalesType salesType;

    InterCompanySalesAtCostPriceHandler intercompanySalesAtCostPrice

    // <GEERU>
    OfficialsServ_RU offSrv;
    // </GEERU>

    RecordSortedList mcrRecordSortedList;
    RecordInsertList recordInsertParmSubLine;
    Set excludeContainerFilters;

    //<GIN>
    RecordInsertList salesParmLineINRecordList;
    //</GIN>
}
```



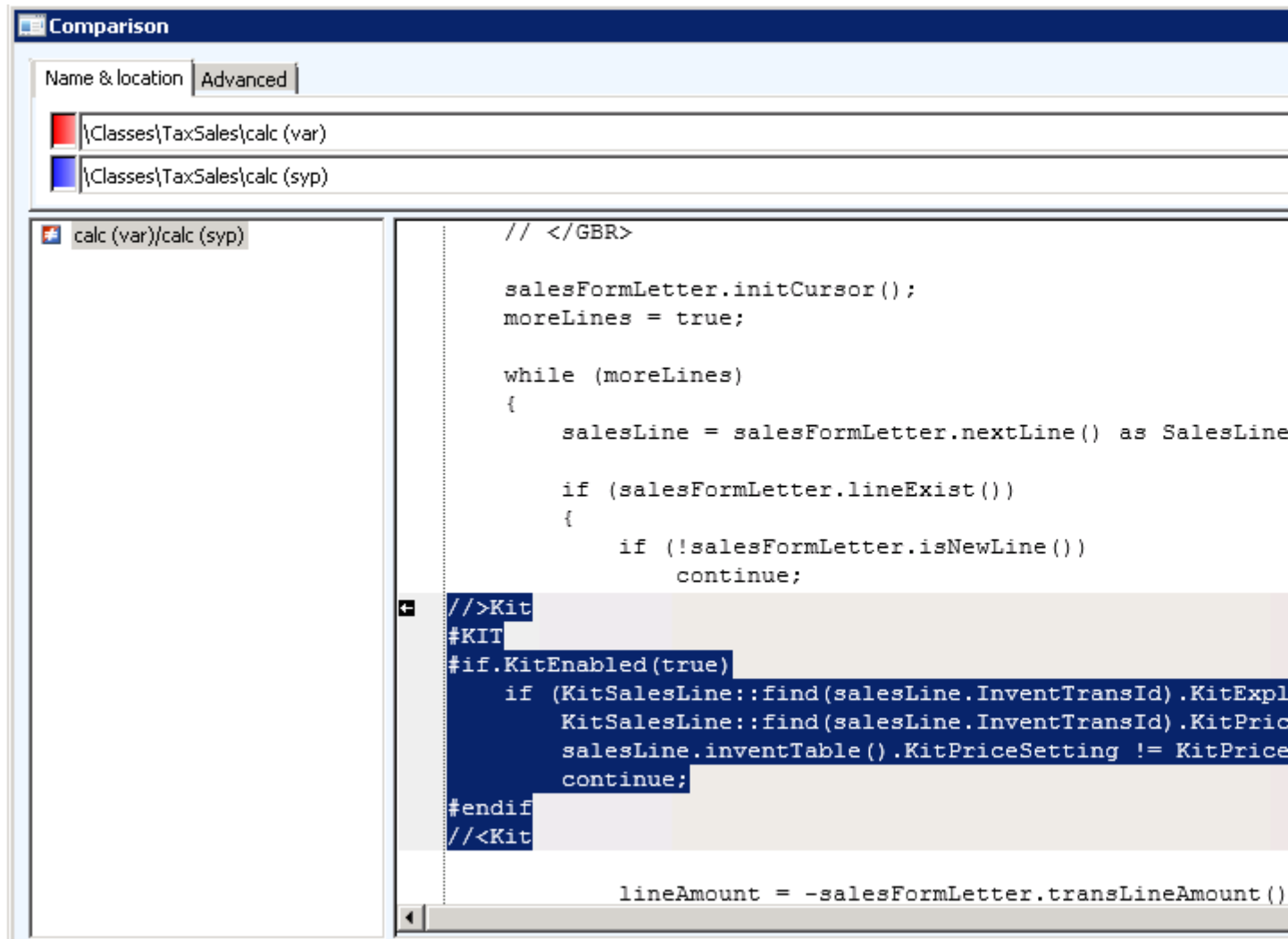
Went back to dev and ran compile on the _KIT project.

Cleaned up errors. Used compare to see differences between var and syp layer, added missing variables back on cus layer. (The utility was built on CU9, not CU13).

For Purchasing related forms, logged-in to VAR layer, switched to "kitting" model, deleted, and removed from project

- PurchCreateFromSalesOrder
- PurchTable

Checked \Classes\TaxSales\calc in TEST. Is not in VAR layer, so it didn't interfere with AvaTax I guess, but I noticed some differences unrelated to Kitting, so deleted from VAR layer and added back in VAR with the just the Kit code changed





```

//>Kit
#KIT
#if.KitEnabled(true)
    if (KitSalesLine::find(salesLine.InventTransId).KitExplosion &&
        KitSalesLine::find(salesLine.InventTransId).KitPriceSetting ==
KitPriceSetting::SumUpPrices &&
        salesLine.inventTable().KitPriceSetting !=
KitPriceSetting::UseKitPrices)
        continue;
#endif
//<Kit


```

Planning to uninstall the model from GOLD and re-install after cleaning up for CU13 differences and such.

Name & location		Advanced
	{Data Dictionary}\Tables\CustInvoiceTrans (var)	
	{Data Dictionary}\Tables\CustInvoiceTrans (syp)	

☒ CustInvoiceTrans (var)/CustInvoice

☒ Methods

 initFromSalesLine

```

// </GEERU>
this.SalesGroup                = salesLine.SalesGroup;
this.QtyPhysical                = 0;
this.PartDelivery              = 0;
this.DefaultDimension          = salesLine.DefaultDimen
this.TransactionCode           = salesLine.TransactionC
this.Transport                 = salesLine.Transport;
this.Port                      = salesLine.Port;
this.StatProcId                = salesLine.StatProcId;
this.LineHeader                = salesLine.LineHeader;
this.DlvDate                   = salesLine.deliveryDate
this.CustomerLineNum           = salesLine.CustomerLine
this.DeliveryType              = salesLine.DeliveryType
this.InventRefType             = salesLine.InventRefTyp
this.InventRefId               = salesLine.InventRefId;
this.InventRefTransId          = salesLine.InventRefTra
this.InterCompanyInventTransId = salesLine.InterCompany

<-->Kit
#KIT
#if.KitEnabled(true)
    if (KitSalesLine::find(salesLine.InventTransId).KitParre
    {
        this.SalesPrice          = 0;
        this.LineAmount          = 0;
        this.LineDisc             = 0;
        this.LinePercent          = 0;
        this.MultiLnDisc          = 0;
        this.MultiLnPercent       = 0;
        this.DiscPercent          = 0;
        this.DiscAmount           = 0;
    }
#endif
//<Kit

    if (pdsIsCWItem(this.ItemId) )
    {
        this.PdsCWQty            = salesLine.PdsCWInventD

```


```


//>Kit
#KIT
#if.KitEnabled(true)
    if (KitSalesLine::find(salesLine.InventTransId).KitParrentInventTransId)
    {
        this.SalesPrice          = 0;
        this.LineAmount          = 0;
        this.LineDisc             = 0;
        this.LinePercent          = 0;
        this.MultiLnDisc          = 0;

```

```
        this.MultiLnPercent    = 0;
        this.DiscPercent       = 0;
        this.DiscAmount        = 0;
    }
#endif
//<Kit
```

Name & location Advanced

 {Data Dictionary}\Tables\SalesLine (var)

 {Data Dictionary}\Tables\SalesLine (syp)

☒ SalesLine (var)/SalesLine (syp)

☒ Field Groups

☒ ItemReference

☒ isKit

☒ Methods

☒ deliveredInventInTotal

☒ initFromPriceDisc

☒ isKit

☒ itemName

☒ kitItemId

☒ lineDeliveryImageResNo

☒ modifyInventDim

☒ onOrder

☒ qtyOnPickNotReserved

☒ reservedOnOrder

☒ reservedOnPick

☒ reservedPhysical

☒ setPriceDisc

☒ lastLineNum

```

/// <summary>
/// Updates the related fields when an inventory dimension is
/// </summary>
/// <param name="_inventDim">
/// The new inventory dimension record.
/// </param>
/// <param name="_dimFieldId">
/// The ID of the modified <c>InventDim</c> field.
/// </param>
/// <param name="_resetPrice">
/// A Boolean value that indicates whether the price agreement
/// </param>
public server void modifyInventDim(
    InventDim          _inventDim,
    FieldId            _dimFieldId,
    boolean            _resetPrice = true)
{
    InventDimGroupSetup    inventDimGroupSetup = InventDimGroupSetup;
    InventDim              inventDimLocal = InventDim;
    SalesReceiptDateRequested    receiptDateRequestedOrig;
    SalesReceiptDateConfirmed    receiptDateConfirmedOrig;
    SalesShippingDateRequested    shippingDateRequestedOrig;
    SalesShippingDateConfirmed    shippingDateConfirmedOrig;
    AgreementHeaderRecId    matchingSalesAgreementHeaderRecId;
    // <GIN>
    SalesLine_IN            salesLine_IN;
    // </GIN>

```

```

//>Kit

```

```

#KIT

```

```

#if.KitEnabled(true)

```

```

    if (_resetPrice &&

```

```

        KitSalesLine::find(this.InventTransId) &&

```

```

        KitSalesLine::find(this.InventTransId).KitParrentInventDimId &&

```

```

        KitSalesLine::find(this.InventTransId).KitPriceSettingId &&

```

```

    {

```

```

        _resetPrice = false;

```

```

    }

```

```

#endif

```



```

//<Kit














```

Comparison

Name & location **Advanced**

	\Data Dictionary\Tables\SalesLine (var)
	\Data Dictionary\Tables\SalesLine (syp)

☒ SalesLine (var)/SalesLine (syp)

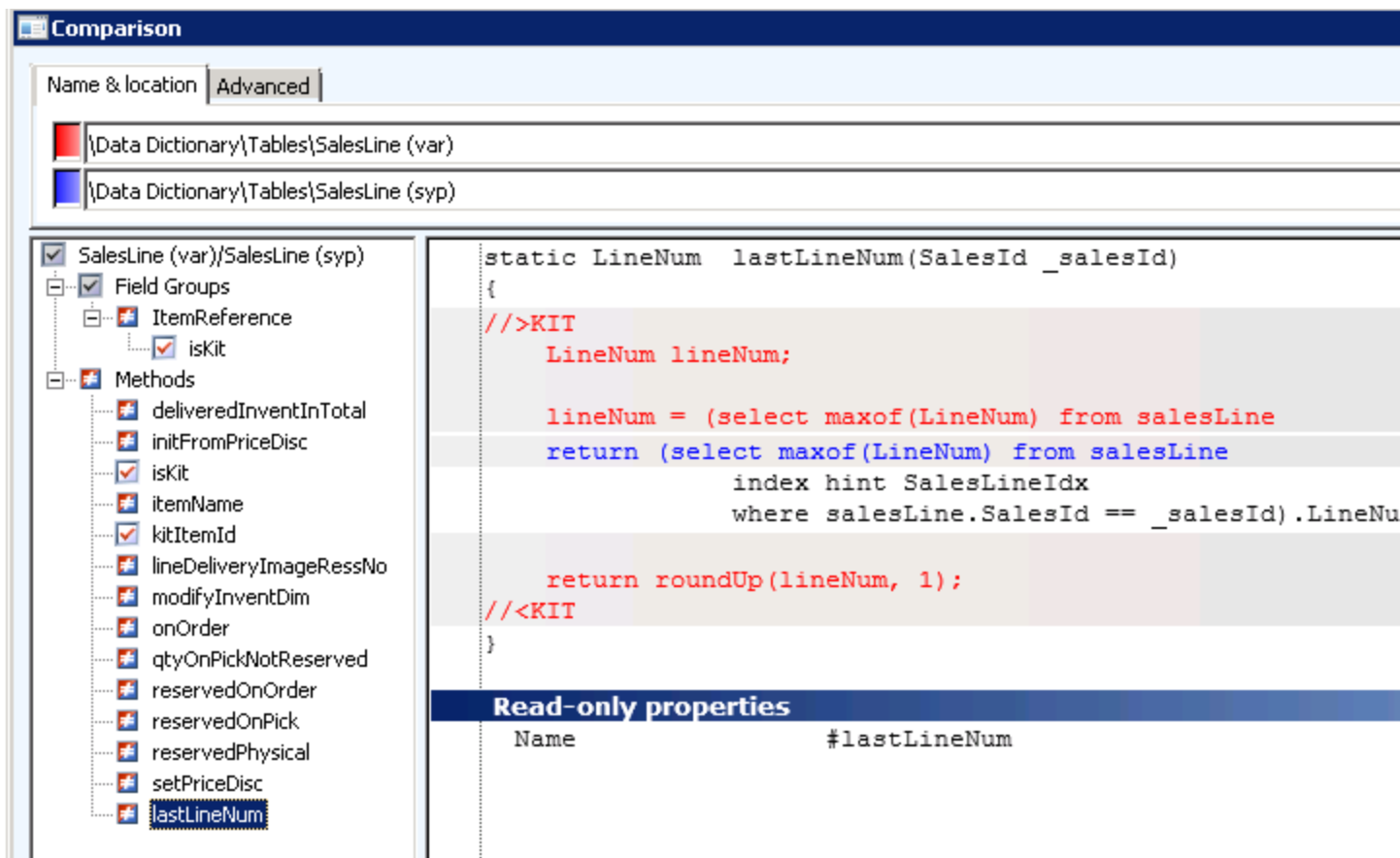
- ☒ Field Groups
 -  ItemReference
 - ☒ isKit
- ☒ Methods
 -  deliveredInventInTotal
 -  initFromPriceDisc
 - ☒ isKit
 -  itemName
 - ☒ kitItemId
 -  lineDeliveryImageResNo
 -  modifyInventDim
 -  onOrder
 -  qtyOnPickNotReserved
 -  reservedOnOrder
 -  reservedOnPick
 -  reservedPhysical
 -  setPriceDisc
 -  lastLineNum

```

server public void setPriceDisc(InventDim _inventDim, boolean
{
    PriceDiscPolicyCheckPolicy checkPolicy = PriceDiscPolic
//>Kit
#KIT
#if .KitEnabled(true)
    KitSalesLine localKitSalesLine = KitSalesLine::f
    if (localKitSalesLine.KitParrentInventTransId != '')
    {
        if (_initDateFields)
        {
            SalesCalcAvailableDlvDates::initDateFields(this,
        }
        if (localKitSalesLine.KitPriceSetting == KitPriceSet
            return;
        }
    }
#endif
//<Kit

```



Some code, like this, is not disabled with the Kit macro...



This one looks odd:


Comparison

Name & location Advanced

	\Data Dictionary\Tables\SalesQuotationLine (var)
	\Data Dictionary\Tables\SalesQuotationLine (syp)

☒ SalesQuotationLine (var)/SalesQuot

☒ Field Groups

 ItemReference

☒ isKit

☒ Methods

☒ initFromBOM
 ☒ initFromPriceDisc
 ☒ isKit
 ☒ itemName
 ☒ kitItemId
 ☒ lineDeliveryImageResNo
 ☒ setAlternativeItem
 ☒ setPriceDisc

```

server void  setPriceDisc(InventDim inventDim, RetailSalesP
{
    PriceDiscPolicyCheckPolicy  checkPolicy = PriceDiscPoli
    MCRSalesQuotationLineOverride mcrSalesQuotationLineOver
    //>Kit
    #KIT
    #if.KitEnabled(true)
        KitSalesLine          localKitSalesLine = KitSalesLine::
        if (localKitSalesLine.KitParrentInventTransId != '' &&
        {
            return;
        }
    #endif
    //<Kit
    if (RetailMCRChannelTable::mcrEnableOrderPriceControlFi
    {
        mcrSalesQuotationLineOverride =  this.mcrSalesQuota
    }

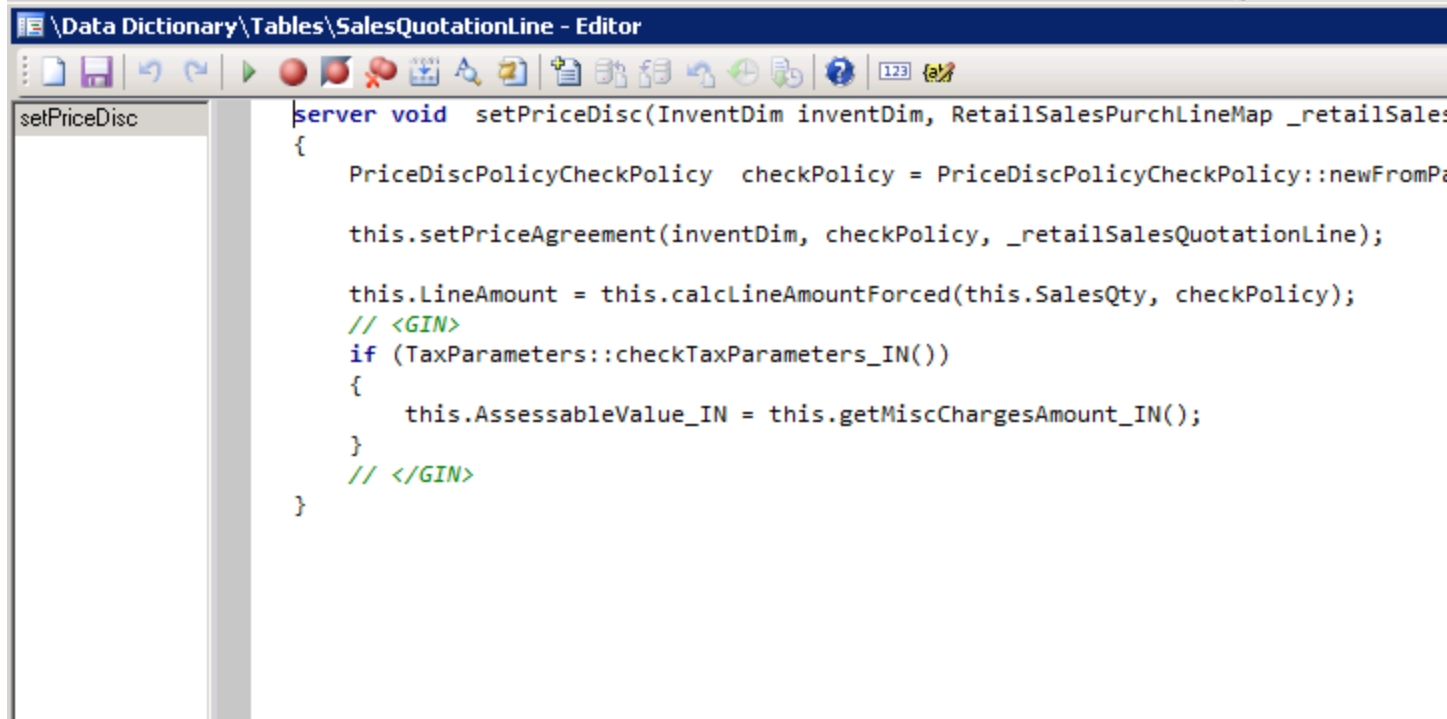
    if (!mcrSalesQuotationLineOverride.PriceOverride)
    {
        this.setPriceAgreement(inventDim, checkPolicy, _ret
    }

    this.setPriceAgreement(inventDim, checkPolicy, _retails

    this.LineAmount = this.calcLineAmountForced(this.SalesQ
    // <GIN>
    if (TaxParameters::checkTaxParameters_IN())

```

Checked code in TEST:



Just adding back to VAR the Kit remarked lines....

```
//>Kit
#KIT
#if.KitEnabled(true)
    KitSalesLine      localKitSalesLine =
KitSalesLine::find(this.InventTransId);
    if (localKitSalesLine.KitParrentInventTransId != '' &&
localKitSalesLine.KitPriceSetting == KitPriceSetting::UseKitPrices)
    {
        return;
    }
#endif
//<Kit
```


Name & location

Advanced



\Classes\AxSalesLine (var)



\Classes\AxSalesLine (syp)

✓ AxSalesLine (var)/AxSalesLine (syp)

setSalesPrice

```
        this.parmSalesPrice (agreementLineQuantityCommitment.  
    }  
    else  
    {  
        this.setPriceDisc_PriceFields ();  
  
        if (this.isPriceDisc_PriceFieldsSet ()  
            || (this.axSalesTable ().isFieldModified (fieldNum  
                && !this.salesLine ().isDropShipmentPOLineInvo  
        )  
        {  
            if (useRetailPricing)  
            {  
                //>Kit  
                #KIT  
                #if.KitEnabled(true)  
                if (KitSalesLine::find(this.salesLine ().InventTra  
                {  
                    exceptionTextFallThrough (); //Do nothing  
                }  
                else if (KitSalesLine::find(this.salesLine ().Inve  
                    KitSalesLine::find(this.salesLine ().Inve  
                    KitSalesLine::find(this.salesLine ().Inve  
                    this.salesLine ().SalesPrice != 0)  
                {  
                    retailCalculator.setPriceOnCurrentLine ();  
                }  
                else  
                {  
                    this.setPriceDisc_PriceFields ();  
  
                    if (this.isPriceDisc_PriceFieldsSet ()  
                        || (this.isShipReceiptDateRequestedFieldsModi  
                            && ( !this.salesLine ().salesTable ().is  
                                || (this.salesLine ().salesTable ().in  
                                    && this.priceDisc_Price ().price  
                                )))  
                    {  
                        exceptionTextFallThrough (); //Do nothing  
                        this.parmSalesPrice (this.priceDisc_Price ().pr  
                    }  
                    else  
                    {  
                        #endif  
                        //<Kit  
                        this.parmSalesPrice (this.priceDisc_Price ().price  
                    }  
                }  
            }  
        }  
    }  
}
```

```

//>Kit
#KIT
#if.KitEnabled(true)
    if
        (KitSalesLine::find(this.salesLine().InventTransId).KitParrentInventTransId
        && KitSalesLine::find(this.salesLine().InventTransId).KitPriceSetting ==
        KitPriceSetting::UseKitPrices)
    {
        exceptionTextFallThrough(); //Do nothing
    }
    else if (KitSalesLine::find(this.salesLine().InventTransId) &&
        KitSalesLine::find(this.salesLine().InventTransId).KitParrentInventTransId ==
        '' &&
        KitSalesLine::find(this.salesLine().InventTransId).KitPriceSetting ==
        KitPriceSetting::SumUpPrices &&
        this.salesLine().SalesPrice != 0)
    {
        exceptionTextFallThrough(); //Do nothing
    }
    else
#endif
//<Kit

```

Fixed...

```

51         if (this.isPriceDisc_PriceFieldsSet()
52             || (this.isShipReceiptDateRequestedFieldsModified()
53                 && ( !this.salesLine().salesTable().isInterCompanyOrder()
54                     || (this.salesLine().salesTable().interCompanyEndpointActionPolicy().Pri
55                         && this.priceDisc_Price().price())
56                     )))
57         {
58             //>Kit
59             #KIT
60             #if.KitEnabled(true)
61                 if (KitSalesLine::find(this.salesLine().InventTransId).KitParrentInventTransId &&
62                     {
63                         exceptionTextFallThrough(); //Do nothing
64                     }
65                 else if (KitSalesLine::find(this.salesLine().InventTransId) &&
66                         KitSalesLine::find(this.salesLine().InventTransId).KitParrentInventTrans
67                         KitSalesLine::find(this.salesLine().InventTransId).KitPriceSetting == Ki
68                         this.salesLine().SalesPrice != 0)
69                     {
70                         exceptionTextFallThrough(); //Do nothing
71                     }
72                 else
73             #endif
74             //<Kit
75
76             this.parmSalesPrice(this.priceDisc_Price().price());
77         }
78     }

```

%

Comparison

Name & location

Advanced

\Classes\RetailCreateLinesFromProductsToAdd (var)

\Classes\RetailCreateLinesFromProductsToAdd (syp)

RetailCreateLinesFromProductsToAc

createKitSalesLine

createProducts

loadKitSalesLine

loadLinesFromCaller

```

case tableNum(RetailSpecialCategoryMember):
    this.loadSpecialCategory();
    break;

case tableNum(InventJournalTrans):
    this.loadCountJournalLines();
    break;

case tableNum(RetailPackageTable):
    this.loadPackageLines();
    break;

//>Kit
#KIT
#if.KitEnabled(true)
    case tableNum(SalesLine):
        this.loadKitSalesLine();
        break;

#endif
//<Kit
    }
}

```



```



//>Kit
#KIT
#if.KitEnabled(true)
    case tableNum(SalesLine):
        this.loadKitSalesLine();
        break;
#endif
//<Kit

```

Comparison

Name & location **Advanced**

	\Classes\RetailSalesOrderCalculator (var)
	\Classes\RetailSalesOrderCalculator (syp)

<input checked="" type="checkbox"/> RetailSalesOrderCalculator (var)/Re  saveSalesOrder  setPriceOnCurrentLine	<pre> /// <summary> /// Set price on the current sales line. /// </summary> public void setPriceOnCurrentLine() { Microsoft.Dynamics.Commerce.Runtime.DataModel.SalesLine CLRObject clrSalesLines, enumeratorSalesLine; str lineId; Price price; <!-->Kit if (KitSalesLine::exist(currentSalesLine.InventTransId)) return; <!--<Kit // Initialize calculation result on RetailSalesTable </pre>
--	--

```

//>Kit
    if (KitSalesLine::exist(currentSalesLine.InventTransId))
        return;
//<Kit

```

\Classes\RetailSalesOrderCalculator - Editor

classDeclaration
setPriceOnCurrentLine

```



1  /// <summary>
2  /// Set price on the current sales line.
3  /// </summary>
4  public void setPriceOnCurrentLine()
5  {
6      Microsoft.Dynamics.Commerce.Runtime.DataModel.SalesLine crtSalesLine;
7      CLRObject clrSalesLines, enumeratorSalesLine;
8      str lineId;
9      Price price;
10
11      RetailSalesOrderCalculatorCache orderCalculatorCache = RetailSalesOrderC
12
13  <!-->Kit
14      if (KitSalesLine::exist(currentSalesLine.InventTransId))
15          return;
16  <!--<Kit
17      if (orderCalculatorCache.shouldCalculatePriceDiscountForLine())
18      {
19          // Initialize calculation result on RetailSalesTable.
20          ttsBegin;
21          retailSalesTable = RetailSalesTable::findSalesTable(salesTable::find
22          if (retailSalesTable.TmpPriceDiscountData)
23          {
24              . . . . .
25          }

```




The screenshot shows the 'Comparison' window in SAP ABAP. The left pane displays the class declaration and methods for 'SalesFormletterParmDataPackingslip'. The right pane shows the code for the 'kitAddParrentSalesLine' method, which includes comments for 'GEERU' and 'GIN' and code for setting 'OfficialServ_RU', 'RecordSortedList', 'RecordInsertList', and 'excludeContainerFilters'.

Comparison

Name & location Advanced

	\Classes\SalesInvoiceDP [var]
	\Classes\SalesInvoiceDP [syp]

☒ SalesInvoiceDP [var]/Sales

-  insertIntoSalesInvoi
-  iteml
-  printDimHis

```

// </GTH>
salesInvoiceTmp.TaxInvoiceSalesId = _custInvoiceJour.Ta
salesInvoiceTmp.ReverseChargeWording_UK = _custInvoiceJ

if (_custInvoiceTrans)
{
    salesInvoiceTmp.LineHeader = this.parmLineHeader();
}

//custInvoiceTrans
salesInvoiceTmp.Name = _custInvoiceTrans.Name;
//>Kit
#KIT
#if.KitEnabled(true)
    salesInvoiceTmp.Name = strRe
#endif
#if.KitEnabled(false)
    salesInvoiceTmp.Name = _custInvoiceTrans.Name;
#endif
//<Kit

if (SysCountryRegionCode::isLegalEntityInCountryRegion(
{
    salesInvoiceTmp.Name = SalesParameters::nameAndOrDe

```

```

//>Kit
#KIT
#if.KitEnabled(true)
    salesInvoiceTmp.Name = strRep(' ',
KitSalesLine::find(_custInvoiceTrans.InventTransId).KitBomLevel) + _custInvoiceTrans.Name;
#endif
#if.KitEnabled(false)
    salesInvoiceTmp.Name = _custInvoiceTrans.Name;
#endif
//<Kit

```

```
104
105     if (_custInvoiceTrans)
106     {
107         salesInvoiceTmp.LineHeader = this.parmlineHeader();
108     }
109
110     ///

100 %


```

☒ ...

AMONGST THE

COSTAMOUNT;

```
void updateInventory()
```



```

: #KIT

```

```
#if.KitEnabled(true)
```

```
if (KitSalesLine::find( inventMovement.transId() ).Kit
```

{

```
updateNow      = salesParmLine.DeliverNow;
```

```
updateNowInvent      = salesParmLine.InventNow;
```

```

cwUpdateNow      = salesParmLine.PdsCWDeliverN

```

```
invoiceUpdatedOnly = updateNow;
```

```
salesQuantityToMatch= 0;
```

```
if ( inventMovement.mustBeRemainControlled())
```

{

```
//Need to find out if enough have been physic
```

```
if (salesParmLine.RemainAfter + inventMoveme
```

1

```
inventMovement.addRemainPhysicalUnit(- i
```

3

```
if (salesParmLine.RemainAfterInvent + invent
```

{

```
inventMovement.addRemainPhysical( - inven
```

}

```
if (min(salesParmLine.salesLine().RemainSales
```

```
inventMovement.addRemainFinancialUnit(mi
```

}

```
localUpdateNow      = updateNow;
```

```
localUpdateNowInvent    = updateNowInvent;
```

```
localInvoiceUpdatedOnly = updateNow;
```

```
//<PMF>
```

```
localCWUpdateNow      = cwUpdateNow;
```

```
localCWInvoiceUpdatedOnly = cwUpdateNow:
```

//</PMF>

return:

1

```
#endif
```

$$1 \leq k \leq K$$

// CREDIT\

```
//>Kit
```

#KIT


```

#if.KitEnabled(true)
    if (KitSalesLine::find(_inventMovement.transId()).KitExplosion)
    {
        updateNow          = salesParmLine.DeliverNow;
        updateNowInvent    = salesParmLine.InventNow;
        cwUpdateNow        = salesParmLine.PdsCWDeliverNow;
        invoiceUpdatedOnly = updateNow;
        salesQuantityToMatch= 0;

        if (_inventMovement.mustBeRemainControlled())
        {
            //Need to find out if enough have been physical updated


            if (salesParmLine.RemainAfter + _inventMovement.remainPhysicalUnit() != 0)
            {
                _inventMovement.addRemainPhysicalUnit(-_inventMovement.remainPhysicalUnit() -
salesParmLine.RemainAfter);
            }


            if (salesParmLine.RemainAfterInvent + _inventMovement.remainPhysical() != 0)
            {
                _inventMovement.addRemainPhysical( -_inventMovement.remainPhysical() -
salesParmLine.RemainAfterInvent);
            }

            if (min(salesParmLine.salesLine().RemainSalesFinancial, updateNow) >= 0)
_ininventMovement.addRemainFinancialUnit(min(salesParmLine.salesLine().RemainSalesFinancial, updateNow) ,
min(salesParmLine.salesLine().RemainInventFinancial, updateNow) );
        }

        localUpdateNow          = updateNow;
        localUpdateNowInvent    = updateNowInvent;
        localInvoiceUpdatedOnly = updateNow;
        ///
        localCWUpdateNow        = cwUpdateNow;
        localCWInvoiceUpdatedOnly = cwUpdateNow;
        ///
        return;
    }
#endif
//<Kit

```

 \Classes\SalesLineType (var)

 \Classes\SalesLineType (syp)

☒ SalesLineType (var)/SalesLineType (

 delete
 insert
 update
 construct

```
/// line, optional.  
/// </param>  
/// <remarks>  
/// When deleting delivery lines, their association to their  
/// <c>SalesDeliverySchedule</c> table. When calling this met  
/// name="updateOrderLineOfDeliverySchedule" /> parameter se  
/// caller is to update the order line. This is typically pe  
/// lines in a single transaction and the order line recalcul  
/// </remarks>  
void delete(Common childBuffer = null, boolean deletePBA = t  
{  
    #OCCRetryCount  
    SalesDeliverySchedule salesDeliverySchedule;  
    ReqEventProcessKanban reqEventProcessKanban;  
    //>Kit  
    #Kit  
    KitSalesLine kitSalesLine;  
    //<Kit  
    // <GEEU>  
    SalesLine linkedSalesLine;
```



```
//>Kit
#Kit
    KitSalesLine    kitSalesLine;
//<Kit
```

```
//>Kit
#KIT
#if.KitEnabled(true)
    kitSalesLine = KitSalesLine::find(salesLine.InventTransId);
    if (kitSalesLine)
    {
        KitExplode_SalesLine::construct(salesLine).deleteKit(kitSalesLine.KitParrentInventTransId != '');
    }
#endif
//<Kit
```

<div> <div></div> <div>\Classes\SalesLineType (var)</div> </div>	<div> <div></div> <div>\Classes\SalesLineType (syp)</div> </div>
<div> <div> <div> <div>✓</div> <div>SalesLineType (var)/SalesLineType (</div> </div> <div> <div>delete</div> <div>insert</div> <div>update</div> <div>construct</div> </div> </div> </div>	<pre> boolean salesOrderEntryStats; InterCompanySkipUpdate interCompanySkipUpdate ReqEventProcessParm reqEventProcessParm = ReqEvent MCRBrokerOrder mcrBrokerOrder; boolean returnReplacemnt; MCRSalesLineDropShipment mcrSalesLineDropShipment; // <GEERU> boolean countryRegion_RU = SysCountryRegionCode::isLeq // </GEERU> //>Kit #KIT #if.KitEnabled(true) KitSalesLine kitSalesLine; KitExplode_SalesLine kitExplode_SalesLine; #endif //<Kit boolean initDateFields = true; </pre>

<div> <div> <div> <div>✓</div> <div>SalesLineType (var)/SalesLineType (</div> </div> <div> <div>delete</div> <div>insert</div> <div>update</div> <div>construct</div> </div> </div> </div>	<pre> ttsabort; throw error("@SYS93835"); } if (interCompanySkipUpdate == InterCompanySkipUpdate::N && (salesLine_Orig.SalesQty != salesLine.SalesQty salesLine_Orig.SalesUnit != salesLine.SalesUnit)) { this.interCompanyCalcDiscInternal(); this.interCompanyCalcDisc(); } //>Kit #if.KitEnabled(true) kitSalesLine = KitSalesLine::find(salesLine.InventTransId if (kitSalesLine) { kitExplode_SalesLine = KitExplode_SalesLine::construct if (kitSalesLine.KitExplodedQty && (salesLine.SalesOtv != salesLine_Orig.SalesOtv </pre>
--	---

```

//>Kit
#KIT
#if.KitEnabled(true)
    KitSalesLine kitSalesLine;
    KitExplode_SalesLine kitExplode_SalesLine;
#endif
//<Kit

```

```

//>Kit
#if.KitEnabled(true)
    kitSalesLine = KitSalesLine::find(salesLine.InventTransId, true);

    if (kitSalesLine)
    {
        kitExplode_SalesLine = KitExplode_SalesLine::construct(salesLine);

        if (kitSalesLine.KitExplodedQty &&
            (salesLine.SalesQty != salesLine_Orig.SalesQty ||
             salesLine.QtyOrdered != salesLine_Orig.QtyOrdered))
        {
            kitSalesLine.KitExplodedQty = kitExplode_SalesLine.updateKit();
        }

        if (!salesLine_Orig ||
            salesLine.SalesQty != salesLine_Orig.SalesQty ||
            salesLine.SalesPrice != salesLine_Orig.SalesPrice ||
            salesLine.MultiLnDisc != salesLine_Orig.MultiLnDisc ||
            salesLine.LineDisc != salesLine_Orig.LineDisc ||
            salesLine.LinePercent != salesLine_Orig.LinePercent ||
            salesLine.MultiLnPercent != salesLine_Orig.MultiLnPercent ||
            salesLine.LineAmount != salesLine_Orig.LineAmount ||
            salesLine.CurrencyCode != salesLine_Orig.CurrencyCode ||
            salesLine.InventDimId != salesLine_Orig.InventDimId
        )

        {
            kitExplode_SalesLine.kitPriceCalcUp();
            kitExplode_SalesLine.kitPriceDiscCalc();
        }

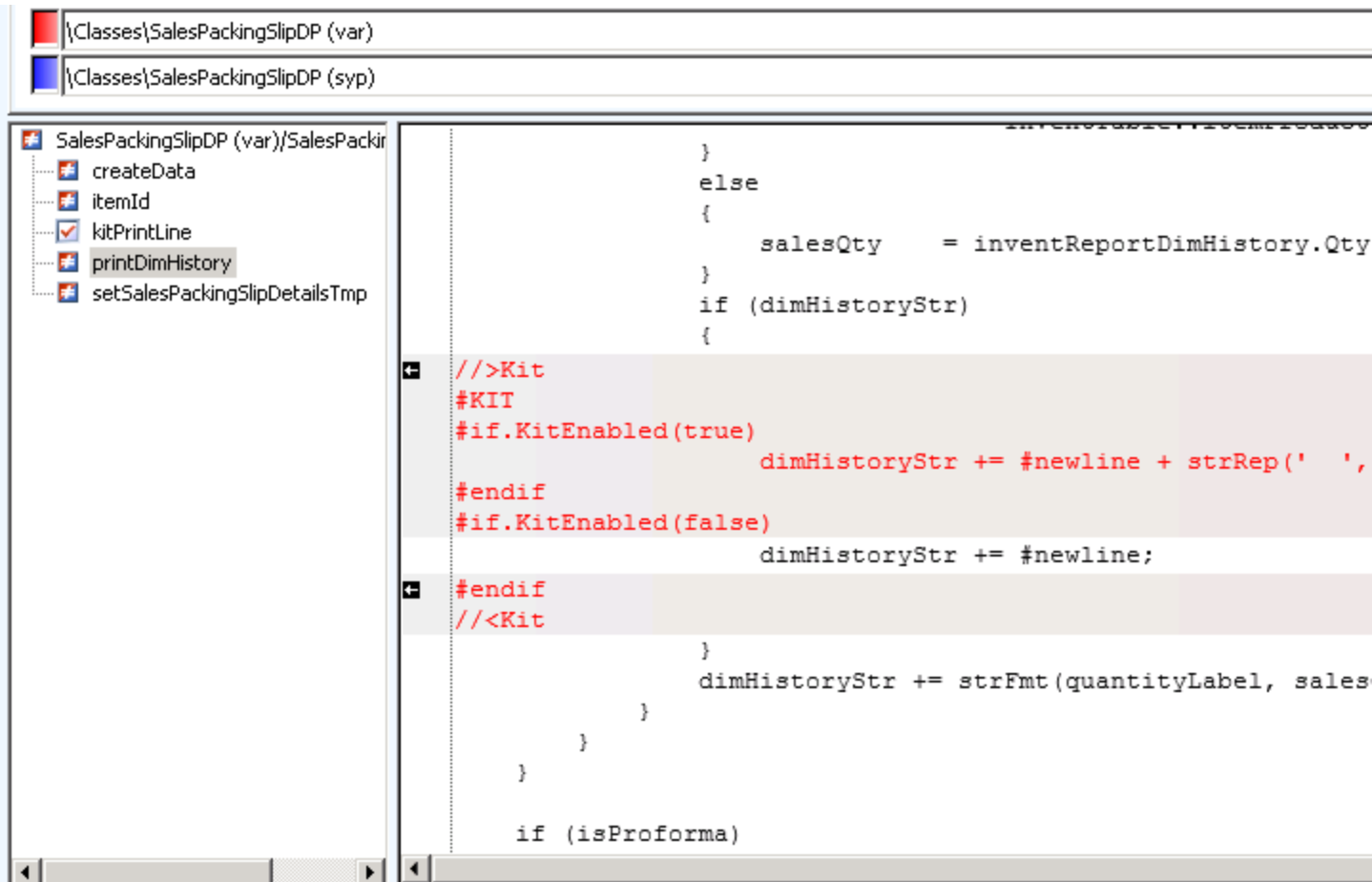
        if (kitSalesLine.KitForceUpdate == NoYes::Yes ||
            salesLine.ShippingDateRequested != salesLine_Orig.ShippingDateRequested ||
            salesLine.ShippingDateConfirmed != salesLine_Orig.ShippingDateConfirmed ||
            salesLine.ReceiptDateRequested != salesLine_Orig.ReceiptDateRequested ||
            salesLine.ReceiptDateConfirmed != salesLine_Orig.ReceiptDateConfirmed ||
            salesLine.DeliveryDateControlType != salesLine_Orig.DeliveryDateControlType)
        {
            kitExplode_SalesLine.kitDeliveryDateCalcUp();
            kitExplode_SalesLine.kitDeliverydates();
            kitSalesLine.forceUpdateDisable();
        }

        if (kitSalesLine.KitParrentInventTransId &&
            (kitSalesLine.salesLine().ShippingDateRequested >
             kitSalesLine.parrentSalesLine().ShippingDateRequested ||
             kitSalesLine.salesLine().ShippingDateConfirmed >
             kitSalesLine.parrentSalesLine().ShippingDateConfirmed ||
             kitSalesLine.salesLine().ReceiptDateRequested >
             kitSalesLine.parrentSalesLine().ReceiptDateRequested ||
             kitSalesLine.salesLine().ReceiptDateConfirmed >
             kitSalesLine.parrentSalesLine().ReceiptDateConfirmed))
        {
            kitExplode_SalesLine.kitDeliveryDateCalcUp();
        }

        if (salesLine.SalesDeliverNow != salesLine_Orig.SalesDeliverNow ||
            salesLine.InventDeliverNow != salesLine_Orig.InventDeliverNow)
        {
            kitExplode_SalesLine.kitDeliverNow();
        }

        if ((salesLine.inventDim().InventLocationId != salesLine_Orig.inventDim().InventLocationId) ||
            (salesLine.inventDim().InventSiteId != salesLine_Orig.inventDim().InventSiteId))
        {
            //Site and warehouse is changed, and any kitting lines must be updated
            kitExplode_SalesLine.kitInventDimChange();
        }
    }

```

```
//>Kit  
#KIT  
#if.KitEnabled(true)  
    dimHistoryStr += #newline + strRep(' ',  
KitSalesLine::find(custPackingSlipTrans.InventTransId).KitBomLevel) ;  
#endif  
#if.KitEnabled(false)  
    dimHistoryStr += #newline;  
#endif  
//<Kit
```

\Classes\SalesPackingSlipJournalPost (syp)

```
[ 36]         _inventMovement.canCommittedCostBeReversed
[ 37]         {
[ 38]             costControlPostingItem= CostControlPostingItem
[ 39]             costControlPostingItem.run(ProjCommittedCostControlPostingItem)
[ 40]         }
+ [ 41]     }
[ 42]
+ [ 43] //>Kit
[ 44] #KIT
[ 45] #if.KitEnabled(true)
[ 46]     if (_inventMovement.kitExplosion())
[ 47]     {
[ 48]         updateNow          = salesParmLine.DeliverNow - salesParmLine.InventNow
[ 49]         updateNowInvent    = salesParmLine.InventNow - salesParmLine.DeliverNow
[ 50]         // <PMF>
[ 51]         cwUpdateNow        = salesParmLine.PdsCWDeliverNow - salesParmLine.PdsCWInventNow
[ 52]         // </PMF>
[ 53]         if (_inventMovement.mustBeRemainControlled())
+ [ 54]         if (whsCancel)
[ 55]         {
+ [ 56]             _inventMovement.addRemainPhysicalUnit(updateNow, updateNowInvent)
[ 57]             _inventMovement.addRemainPhysical(updateNow, updateNowInvent)
[ 58]             _inventMovement.addRemainFinancialUnit(-updateNow, -updateNowInvent)
[ 59]             _inventMovement.pdsCWAddRemainPhysical(cwUpdateNow, cwUpdateNowInvent)
+ [ 60]             _inventMovement.pdsCWAddRemainFinancial(-cwUpdateNow, -cwUpdateNowInvent)
+ [ 61]             this.pickCancelledTransactions(_inventMovement, _inventPhys)
[ 62]         }
+ [ 63]     return;
[ 64]     }
+ [ 65] #endif
[ 66] //<Kit
[ 67]
```

```
//>Kit
#KIT
#if.KitEnabled(true)
    if (_inventMovement.kitExplosion())
    {
        updateNow          = salesParmLine.DeliverNow - salesParmLine.PreviousDeliverNow ;
        updateNowInvent    = salesParmLine.InventNow - salesParmLine.PreviousInventNow;
        // <PMF>
        cwUpdateNow        = salesParmLine.PdsCWDeliverNow - salesParmLine.PdsCWPreviousDeliverNow;
        // </PMF>
        if (_inventMovement.mustBeRemainControlled())
        {
            _inventMovement.addRemainPhysicalUnit(updateNow);
            _inventMovement.addRemainPhysical(updateNowInvent);
            _inventMovement.addRemainFinancialUnit(-updateNow, -updateNowInvent);
            _inventMovement.pdsCWAddRemainPhysical(cwUpdateNow);
            _inventMovement.pdsCWAddRemainFinancial(-cwUpdateNow);
        }
    }
return;
```



```

    }
#endif
//<Kit

```

```

80         costControlPostingItem= CostControlPosting::newCostControlPostingItem(_inventMovement);
81         costControlPostingItem.run(ProjCommittedCostUpdateAction::Delete);
82     }
83 //>Kit
84 #KIT
85 #if.KitEnabled(true)
86     if (_inventMovement.kitExplosion())
87     {
88         updateNow      = salesParmLine.DeliverNow - salesParmLine.PreviousDeliverNow ;
89         updateNowInvent = salesParmLine.InventNow - salesParmLine.PreviousInventNow;
90         // <PMF>
91         cwUpdateNow     = salesParmLine.PdsCwDeliverNow - salesParmLine.PdsCwPreviousDeliverNow;
92         // </PMF>
93         if (_inventMovement.mustBeRemainControlled())
94         {
95             _inventMovement.addRemainPhysicalUnit(updateNow);
96             _inventMovement.addRemainPhysical(updateNowInvent);
97             _inventMovement.addRemainFinancialUnit(-updateNow, -updateNowInvent);
98             _inventMovement.pdsCwAddRemainPhysical(cwUpdateNow);
99             _inventMovement.pdsCwAddRemainFinancial(-cwUpdateNow);
100         }
101         return;
102     }
103 #endif
104 //<Kit
105     if (whsCancel)
106     {
107         this.pickCancelledTransactions(_inventMovement,
108                                     inventPhysicalOnhandChangingHandler.parmInventPhysical);
109     }
110 }
111
112     localCustPackingSlipTrans = this.journalLine().data();


```

```
|Classes\SalesQuantity_PackingSlip (var)
```

\Classes\SalesQuantity_PackingSlip (syp)

☒ SalesQuantity_PackingSlip (var)/Sale

calcQtyInvent

..  calcQtySales

getQtyPicked

... pdsCw/CalcQtyInvent

```
[ 54]         }
[ 55]         break;
[ 56]
[ 57]         case SalesUpdate::PickingList,
[ 58]             SalesUpdate::PickingListAndNonInventTransIdSum:
[ 59]             inventTransIdSum = InventTransIdSum;
[ 60]             inventTransIdSum.parmInventDimCriteria();
[ 61]             inventTransIdSum.parmInventDimParameters();
[ 62]             //>Kit
[ 63]             #KIT
[ 64]             #if.KitEnabled(true)
[ 65]                 if (KitSalesLine::find(salesLine.InventTransIdSum))
[ 66]                 {
[ 67]                     qtyUpdate = -salesLine.pickedInTotal();
[ 68]                 }
[ 69]             else
[ 70]             #endif
[ 71]             //<Kit
[ 72]             qtyUpdate = UnitOfMeasureConverter::
[ 73]                 -inventTransIdSum.picked(),
[ 74]                 UnitOfMeasure::unitOfMeasureInventory,
[ 75]                 UnitOfMeasure::unitOfMeasureInventory;
```

```
//>Kit
```

#KIT

```
#if.KitEnabled(true)
```

```
if (KitSalesLine::find(salesLine.InventTransId))
```

 $\{$

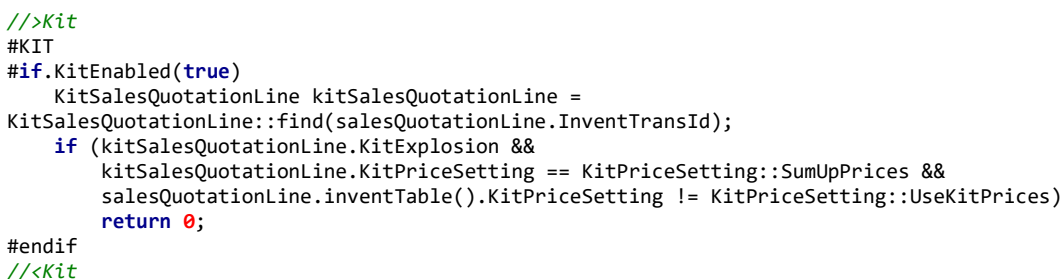
```
qtyUpdate = -salesLine.pickedInTotalSalesUnit();
```

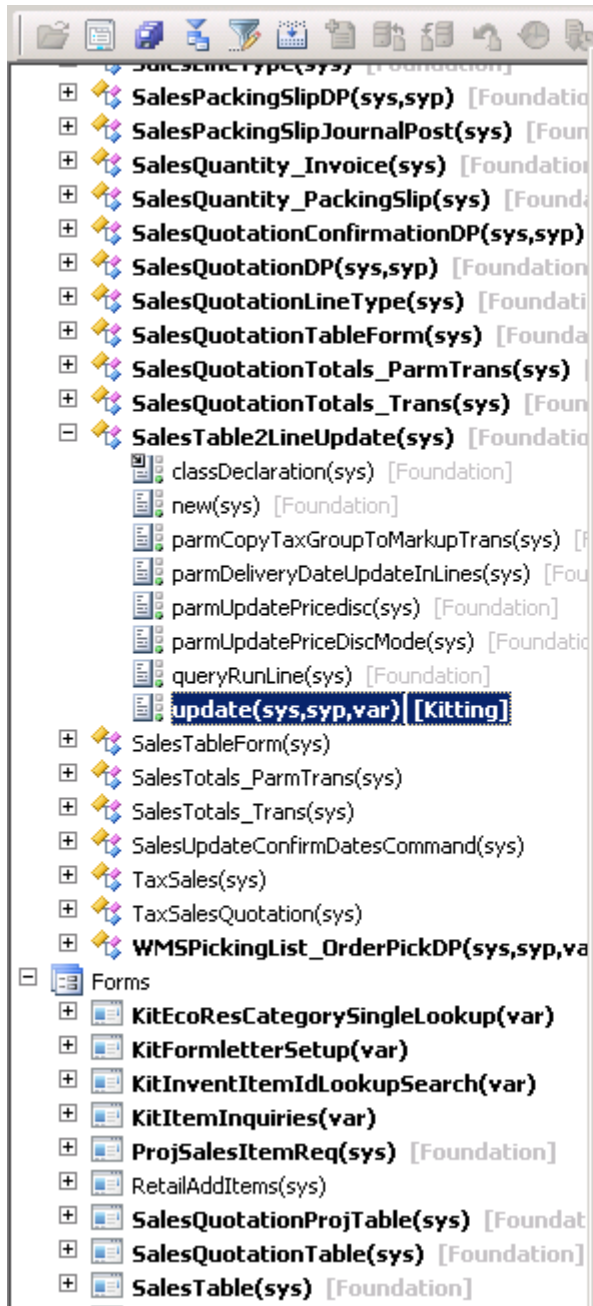
}

```
else
```

```
#endif
```

```
//<Kit
```





```

[ 110]         if (queryRun.changed(tableNum(SalesLine)))
[ 111]         {
[ 112]             salesLine = queryRun.get(tableNum(SalesLine));
[ 113]             salesLine = salesLine::findRecId(salesLine.RecId,true);
[ 114]             if (salesLine)
[ 115]             {
[ 116]                 inventMovement = InventMovement::construct(salesLine);

```

```

[ 184]         }
[ 185]     }
[ 186]     salesLine.SalesPurchLine::setPriceDiscFixedPolicy(salesLine,
[ 187]     if (/updateTaxGroupOnLines)

```

Deleted from VAR and from project.

\Classes\SalesTotals_ParmTrans (var)

\Classes\SalesTotals_ParmTrans (isv)

SalesTotals_ParmTrans (var)/SalesT

- lineAmount
- lineCostPrice
- lineDiscAmount
- pdsCWQtyRemain
- pdsCWQtyUnit
- qtyInvent
- qtyInventRemain
- qtyUnit

```

[ 1] protected LineAmount lineAmount(SalesQty _salesQty, Tr
[ 2] {
[ 3] //>Kit
[ 4] #KIT
[ 5] #if.KitEnabled(true)
[ 6]
[ 7]     if (KitSalesLine::find(salesParmLine.InventTransId)
[ 8]         KitSalesLine::find(salesParmLine.InventTransId
[ 9]     {
[10]         if (salesParmLine.inventTable().KitPriceSetting
[11]             return 0;
[12]     }
[13] #endif
[14] //<Kit
[15]     if (skipTaxCalculations)
[16]     {
[17]         return salesParmLine.effectiveLineAmount();
[18]     }
[19]     else
[20]     {
[21]         return salesParmLine.lineAmountExclTax(_transD
[ 9]         return salesParmLine.lineAmountExclTax(_transD

```

```

//>Kit
#KIT
#if.KitEnabled(true)

    if (KitSalesLine::find(salesParmLine.InventTransId).KitExplosion &&
        KitSalesLine::find(salesParmLine.InventTransId).KitPriceSetting == KitPriceSetting::SumUpPrices)
    {
        if (salesParmLine.inventTable().KitPriceSetting != KitPriceSetting::UseKitPrices)
            return 0;
    }
#endif
//<Kit

```

☒

#KIT

```
#if.
```

KitSalesLine kit

```
if (kitSalesLine.KitExplosion &&
```

kitSalesLine.KitPriceSetting

```
salesLine.inventTable().KitPriceSetting != KitPriceSetting::Use
```

```
return 0;
```

```
//<Kit
```

\Classes\SalesUpdateConfirmDatesCommand (var)

\Classes\SalesUpdateConfirmDatesCommand (syp)

SalesUpdateConfirmDatesCommand
✓ kitUpdateSalesLine
run
selectSalesLine

```
[ 22]         salesLine = this.selectSalesLine();  
[ 21]         salesLine = this.selectSalesLine(true);  
[ 23]         while (salesLine.RecId)  
[ 24]         {  
[ 25]             //>Kit  
[ 26]             #KIT  
[ 27]             localSalesLine = salesLine::findRecId(salesLine.RecId);  
[ 28]             setPrefix(this.prefixTextSalesLine(localSalesLine));  
[ 24]             setPrefix(this.prefixTextSalesLine(salesLine));  
[ 29]  
[ 30]             // If it allows the new confirmed dates, o  
[ 31]             if (salesLine.AllowNewConfirmedDates)  
[ 32]             {  
[ 33]                 localSalesLine.ShippingDateConfirmed = date  
[ 34]                 localSalesLine.ReceiptDateConfirmed = date  
[ 29]                 salesLine.ShippingDateConfirmed = date  
[ 30]                 salesLine.ReceiptDateConfirmed = date  
[ 35]             }  
[ 36]  
[ 37]             if (this.setConfirmedDates(localSalesLine, salesLine))  
[ 33]             if (this.setConfirmedDates(salesLine, salesLine))  
[ 38]             {  
[ 39]                 this.updateSalesLine(localSalesLine);  
[ 40]                 localSalesLine = salesLine::findRecId(salesLine.RecId);  
[ 41]                 if (! this.isRequestedDatesExceeded(localSalesLine))  
[ 35]                 this.updateSalesLine(salesLine);  
[ 36]  
[ 37]                 if (! this.isRequestedDatesExceeded(salesLine))  
[ 42]                 {  
[ 43]                     numOfConfirmedSalesLines++;  
[ 44]                 }  
[ 45]             //<Kit  
[ 41]         }  
[ 42]         else  
[ 43]         {  
[ 44]             if (salesLine.ShippingDateConfirmed && salesLine.ReceiptDateConfirmed)  
[ 45]             {  
[ 46]                 numOfConfirmedSalesLines++;  
[ 47]             }  
[ 47]         }  
[ 47]     }  
[ 47] }
```

```

[ 64]
[ 65]         strategy = this.strategyToConfirmSalesLines();
[ 66]
[ 67]         ttsbegin;
[ 68] //>Kit
[ 69] #KIT
[ 70] #if.KitEnabled(true)
[ 71]         this.kitUpdateSalesLine();
[ 72] #endif
[ 73] //<Kit
[ 74]
[ 75]         if (!allowMultipleShipDates)
[ 76]         {
[ 77]                 strategy.parmSkipConflictHandling(true);
[ 78]                 strategy.parmNetCalculateConfirmedDates(false);

```

salesLine localSalesLine;

```

//>Kit
#KIT

localSalesLine = salesLine::findRecId(salesLine.RecId,true);
setPrefix(this.prefixTextSalesLine(localSalesLine));

// If it allows the new confirmed dates, clear the old values and recalculate them.
if (salesLineAllowNewConfirmedDates)
{
    localSalesLine.ShippingDateConfirmed = dateNull();
    localSalesLine.ReceiptDateConfirmed = dateNull();
}

if (this.setConfirmedDates(localSalesLine,strategy))
{
    this.updateSalesLine(localSalesLine);
    localSalesLine = salesLine::findRecId(salesLine.RecId);
    if (! this.isRequestedDatesExceeded(localSalesLine))
    {
        numOfConfirmedSalesLines++;
    }
}

//<Kit

```

```



//>Kit
#KIT
#if.KitEnabled(true)
    this.kitUpdateSalesLine();
#endif
//<Kit


```

Only included section with “if.kiteneabled...”

Comparison

Name & location **Advanced**

	\Classes\TaxSalesQuotation (var)
	\Classes\TaxSalesQuotation (syp)

☒ TaxSalesQuotation (var)/TaxSalesQ
 calc

```

[ 32]     salesQuotationFormLetter.setCursor();
[ 33]
[ 34]     moreLines = true;
[ 35]     while (moreLines)
[ 36]     {
[ 37]         salesQuotationLine = salesQuotationFormLetter;
[ 38]
[ 39]         if (salesQuotationLine)
[ 40]         {
[ 41]             if (SalesQuotationTotals_Trans::canBeIn
[ 42]             {
[ 43] //>Kit
[ 44] #KIT
[ 45] #if.KitEnabled(true)
[ 46]     if (KitSalesQuotationLine::find(salesQuotationLi
[ 47]         KitSalesQuotationLine::find(salesQuotationLi
[ 48]         salesQuotationLine.inventTable().KitPriceSet
[ 49]         continue;
[ 50] #endif
[ 51] //<Kit
[ 52]
[ 52]     this.configureTaxForSalesQuotationLi
  
```

```

//>Kit
#KIT
#if.KitEnabled(true)
    if (KitSalesQuotationLine::find(salesQuotationLine.InventTransId).KitExplosion &&
        KitSalesQuotationLine::find(salesQuotationLine.InventTransId).KitPriceSetting ==
KitPriceSetting::SumUpPrices &&
        salesQuotationLine.inventTable().KitPriceSetting != KitPriceSetting::UseKitPrices)
        continue;
#endif
//<Kit
  
```

Forms\ProjSalesItemReq (var)

Forms\ProjSalesItemReq (syp)

ProjSalesItemReq (var)/ProjSalesItemReq (syp)

Methods

classDeclaration
doReFreshDatasource
init

Data Sources

SalesLine

Methods

lineDeliveryTypeImage
kitItemId
displayOption
executeQuery
create
active
validateWrite

InventDim

Fields

InventSiteId

Methods

modified

1] public void modified()

2] {

3] super();

4]

5] if (inventDim.inventDimId

6] {

7] salesLine.modifyInventDim(inv

8] }

9] }

Added code back to this method.

```

public void modified()
{
    super();

    if (inventDim.inventDimId)
    {
        salesLine.modifyInventDim(inventDim, fieldNum(InventDim, InventLocationId));
    }
}

```


\Forms\SalesQuotationTable (var)	
\Forms\SalesQuotationTable (syp)	
<div> <div> SalesQuotationTable (var)/SalesQuo </div> <div> <div>Methods</div> <div> classDeclaration doRefresh init isCalcTax </div> </div> <div> Data Sources </div> <div> Designs </div> </div>	<pre> [126] // </GDK> [127] logisticsPostalAddressHeader_ds.addFieldToSele [128] logisticsPostalAddressHeader_ds.addFieldToSele [129] logisticsPostalAddressHeader_ds.addFieldToSele [130] logisticsPostalAddressLine_ds.addFieldToSelect [131] logisticsPostalAddressLine_ds.addFieldToSelect [132] logisticsPostalAddressLine_ds.addFieldToSelect [133] //>Kit [134] #KIT [135] #if.KitEnabled(true) [136] kitShowLinesBtn.visible(true); [137] #endif [138] //<Kit [139] [140] if (RetailSMB::IsRetailSMBEnabled()) </pre>

```

//>Kit
#KIT
#if.KitEnabled(true)
    kitShowLinesBtn.visible(true);
#endif
//<Kit

```

\Forms\SalesQuotationTable (var)	
\Forms\SalesQuotationTable (syp)	
<div> <div> SalesQuotationTable (var)/SalesQuo </div> <div> <div>Methods</div> <div> classDeclaration doRefresh init isCalcTax </div> </div> <div> Data Sources </div> <div> <div> SalesQuotationTable </div> <div> <div>Methods</div> <div> enableRetailFields executeQuery create write </div> </div> </div> <div> Fields </div> <div> SalesQuotationLine </div> <div> RetailSalesQuotationTable </div> <div> Designs </div> </div>	<pre> [1] /// <summary> [2] /// Makes manual discount percentage field visible an [3] /// </summary> [4] /// <param name="_enableDisableDiscountFields"> [5] /// The flag to enable disable specific fields. [6] /// </param> [7] /// <remarks> [8] /// Use for retail quote. [9] /// </remarks> [10] private void enableRetailFields(boolean _enableDisabl [11] { [12] salesQuotationLine_ds.object(fieldNum(SalesQuotat [13] salesQuotationLine_ds.object(fieldNum(SalesQuotat [14] [15] retailSalesQuotationLine_ds.object(fieldNum(Retai [16] RetailGroupHeaderView.visible(_enableDisableDisco [17] [16] } </pre>








Line isn't missing...??

```

10 private void enableRetailFields(boolean _enableDisableDiscountFi
11 {
12     salesQuotationLine_ds.object(fieldNum(SalesQuotationLine, Li
13     salesQuotationLine_ds.object(fieldNum(SalesQuotationLine, Li
14
15     retailSalesQuotationLine_ds.object(fieldNum(RetailSalesQuota
16     RetailGroupHeaderView.visible(_enableDisableDiscountFields);
17
18 }

```

SalesTable

Compiler output		
 3 Errors	 4 Warnings	 0 Best practices
 0 Tasks	Reset	Edit Setup ▾ >>
	Description	Path
	Unknown source for data method getItemBrand_MX. The method does not exist in the data source or table.	\Forms\SalesTable\Design
	Variable retailSalesTable has not been declared.	\Forms\SalesTable\Meth
	Variable salesTableCaller has not been declared.	\Forms\SalesTable\Meth

Comparison

Name & location

Advanced

Forms\SalesTable (var)

Forms\SalesTable (isv)

SalesTable (var)/SalesTable (isv)

Methods

classDeclaration

doRefresh

init

Data Sources

Designs

[174] boolean mcrProcessGiftCard

[175] SysExtensionSerializerFormRunHelper formRunHelper

[176] MCRSalesOrderGiftCardPolicyHelper giftCardPolicyC

[176] boolean isResearching;

[177]

[178] UnknownNoYes performIntercompa

[179] </>Kit

[180] #KIT

[181] #if.KitEnabled(true)

[182] boolean kitShowLines;

[183] QueryBuildDataSource kitQDS;

[184] QueryBuildRange kitQR;

[185] #endif

[186] </>Kit

[187] boolean inDeleteMarked;

[188] // <GMY>

[189] boolean isIsoMY;

[190] boolean isSalesLineWE

[191] boolean isWrnPrintedM

[192] CustInvoiceJourLookupController_MY custInvoiceJo

[193] // </GMY>

[186] boolean isRefreshing;

[187] SalesTable salesTableCaller;

[184] 1

```

//>Kit
#KIT
#if.KitEnabled(true)
    boolean
        QueryBuildDataSource
        QueryBuildRange
    kitShowLines;
    kitQDS;
    kitQR;
#endif
</>Kit

```

Put at the bottom instead of the middle.

\Forms\SalesTable (syp)

☒ SalesTable (var)/SalesTable (syp)☒ Methods

doRefresh

 init

  Data Sources

☐...☒ Designs

```
[ 1] void doRefresh(boolean _lineRefreshNeeded = false)
[ 2] {
[ 3]     ///
```

Forms\SalesTable (var)

\Forms\SalesTable (syp)

☒ SalesTable (var)/SalesTable (syp)

 Methods

.... doRefresh

```
....  init
```

  Data Sources

... Designs

```
[ 18] // update header controls
[ 19] element.mcrSetHeaderControls();
[ 20] if (_lineRefreshNeeded)
[ 21] {
[ 22]     salesLine_ds.reread();
[ 23]     salesLine_ds.refresh();
[ 24] element.doRefreshLineActions();
[ 25] }
[ 26] else
[ 27] {
[ 28]     // With addition of executeQuery (see below)
[ 29]     // salesLine_ds.reread() fails if a new, unsaved
[ 30]     // Changed to perform a new query, which will
[ 31]     // The new query will cause line controls to
[ 32]     salesLine_ds.executeQuery();
[ 33] }
[ 34] //>Kit
[ 35] #if.KitEnabled(true)
[ 36]     salesline_ds.findRecord(SalesLine::findRecId(sale
[ 37] #endif
[ 38] //<Kit
[ 39] }
```

```

//>Kit
#KIT
#if.KitEnabled(true)
    RecId    salesLineRecid = salesline.RecId;
#endif
//<Kit

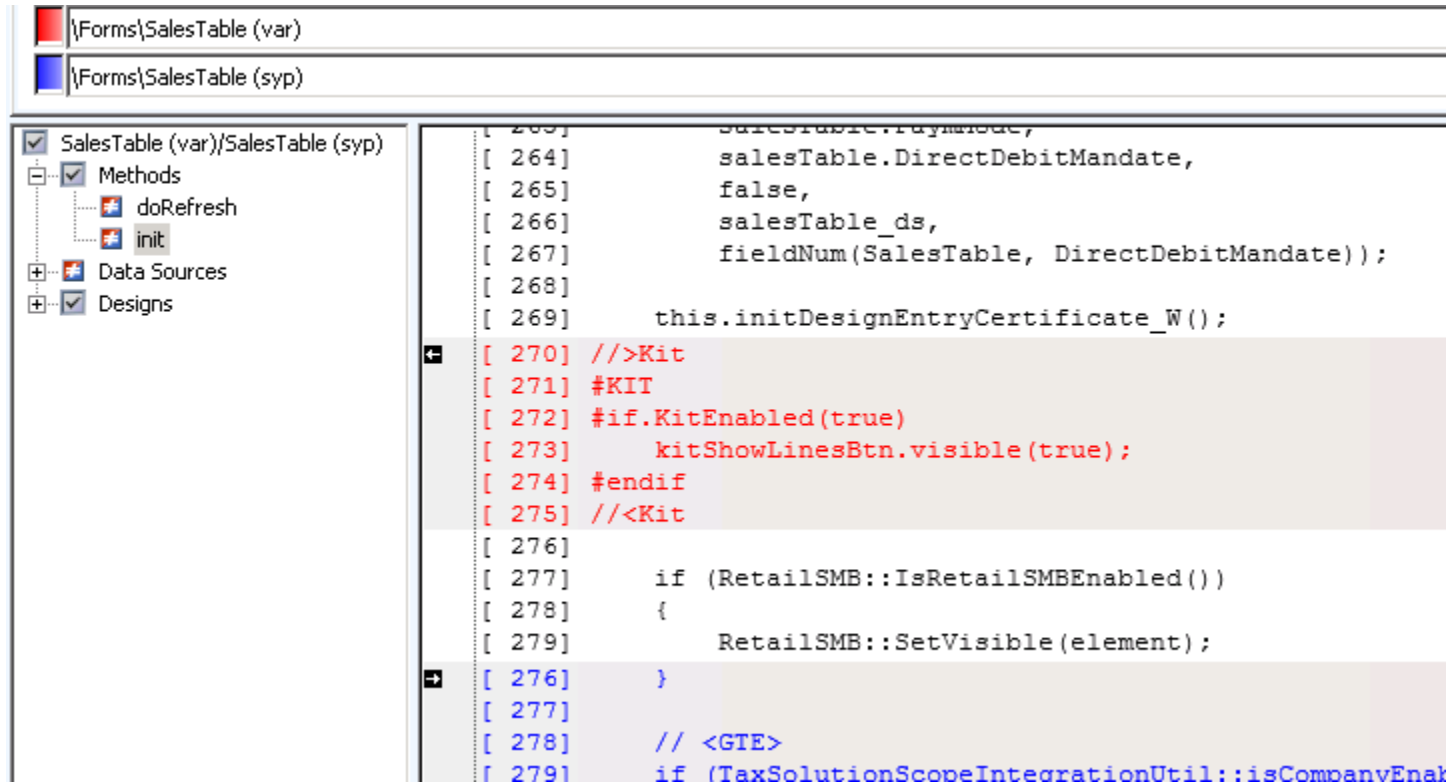
```



```

//>Kit
#if.KitEnabled(true)
    salesline_ds.findRecord(SalesLine::findRecId(salesLineRecid));
#endif
//<Kit

```



```

//>Kit
#KIT
#if.KitEnabled(true)
    kitShowLinesBtn.visible(true);
#endif
//<Kit

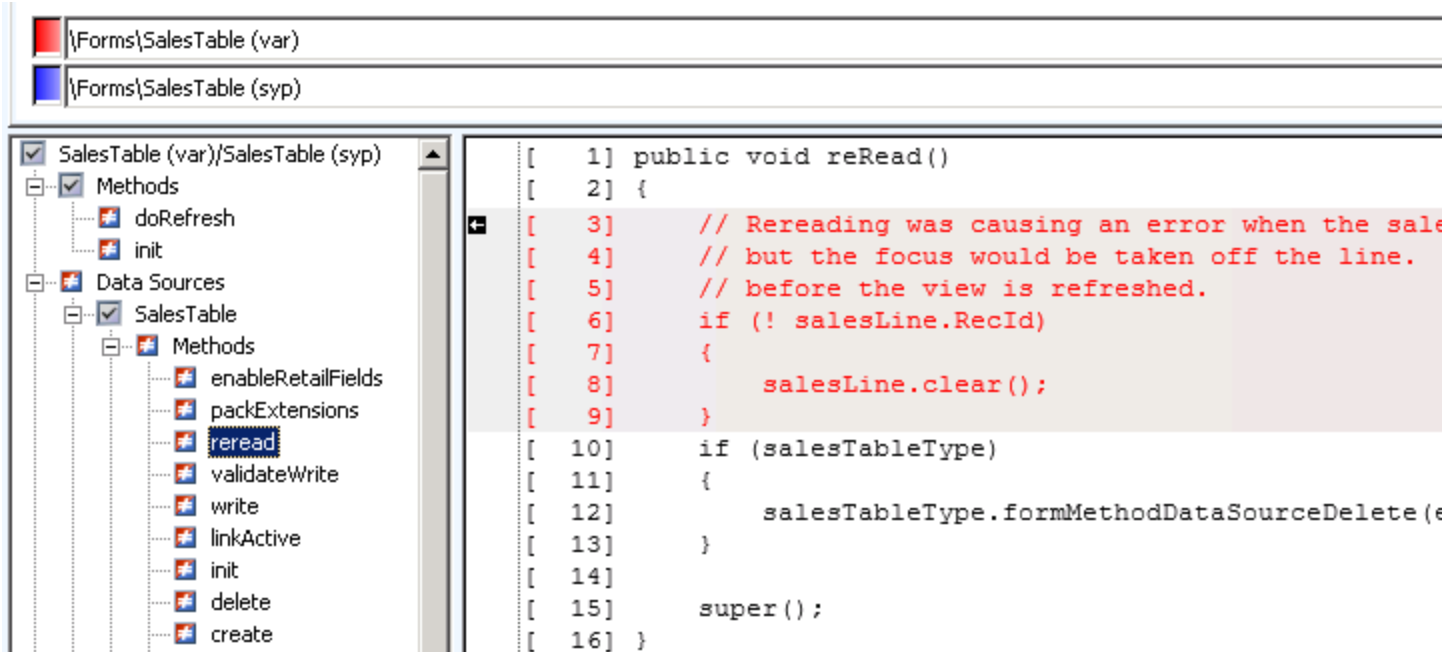
```

Deleted methods in \Data Sources\SalesTable that don't show any Kitting code:

enableRetailFields

packExtensions

reread



validateWrite

write

linkActive

init

delete

active

leave

showDimAccount

Forms\SalesTable (var)

Forms\SalesTable (syp)

SalesTable (var)/SalesTable (syp)

Methods

doRefresh

init

Data Sources

SalesTable

Methods

enableRetailFields

packExtensions

reread

validateWrite

write

linkActive

init

delete

create

active

leave

showDimAccount

updateLineLevelChar

executeQuery

research

Fields

CustAccount

Methods

modified

DiscPercent

Methods

modified

ECommerceOperator_

Methods

lookup

modified

ECommerceOperatorC

Methods

lookupReferen

ECommerceSale_IN

Methods

modified

EndDisc

Methods

validate

modified

LineDisc

Methods

validate

```

[ 1] /// <summary>
[ 2] /// Makes manual discount percentage field visible
[ 3] /// </summary>
[ 4] /// <param name="_enableRetailDiscountFields">
[ 5] /// The flag to enable disable specific fields.
[ 6] /// </param>
[ 7] /// <remarks>
[ 8] /// Use for retail orders.
[ 9] /// </remarks>
[10] private void enableRetailFields(boolean _enableRet
[11] {
[12]     salesLine_ds.object(fieldNum(SalesLine, LineD
[13]     salesLine_ds.object(fieldNum(SalesLine, LineP
[14]
[15]     retailSalesLine_ds.object(fieldNum(RetailSale
[16]     // Disable linemanualdiscountpercentage for re
[17]     // Hide linemanualdiscountpercentage if it is
[18]     if (_enableRetailDiscountFields)
[19]     {
[20]         retailSalesLine_ds.object(fieldNum(RetailS
[21]     }
[22]     else
[23]     {
[24]         retailSalesLine_ds.object(fieldNum(RetailS
[25]     }
[26]
[27]     buttonRetailRecalculator.enabled(_enableRetail
[28]     salesTable_ds.object(fieldNum(SalesTable, Disc
[29]     retailSalesTable_ds.object(fieldNum(RetailSale
[30]     retailSalesLine_ds.object(fieldNum(RetailSales
[31]     buttonRetailRecalculator.enabled(_enableRetail
[32]     buttonGroupAffiliations.enabled(_enableRetailD
[33]     retailGroupHeaderView.visible(_enableRetailDis
[34] }

```

But, that doesn't work... data source is completely removed.

<https://community.dynamics.com/ax/f/microsoft-dynamics-ax-forum/111081/deleting-one-form-element-from-model>

“If you made any change to form data sources then all data source methods are copied to you current layer and cannot be revert back to SYS without undoing all of your changes in form data sources.”

So will have to find all the Kitting changes in the form, document them, delete from VAR layer, and add them back.

SalesTable Modifications for DAXKitting

Method
\classDeclaration
\doRefresh
\init
\Forms\SalesTable\Data Sources\SalesLine\Methods\getItemBrand_MX

\Data Sources\SalesLine\Methods\kitItemId

\Data Sources\SalesLine\Methods\create

\Data Sources\SalesLine\Methods\executeQuery

\Data Sources\SalesLine\Methods\active

\Data Sources\SalesLine\Methods\displayOption

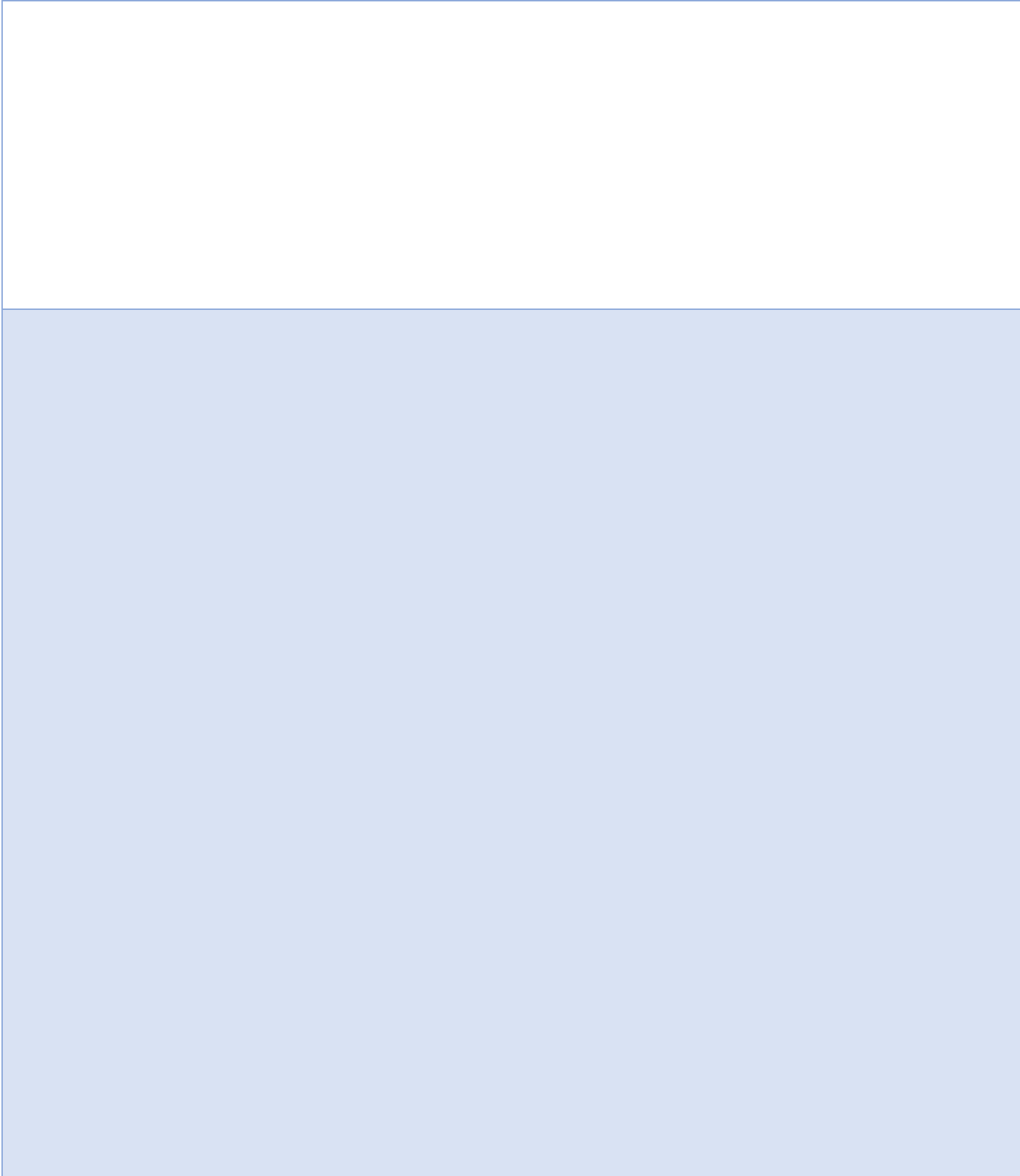
\Data Sources\SalesLine\Methods\validateWrite

\Data Sources\SalesLine\Fields\Itemid\Methods\lookup

\Forms\SalesTable\Designs\Design\[ActionPane:ActionPaneHeader]\[ActionPaneTab:Sell]\[ButtonGroup:SellCalcula

\Forms\SalesTable\Designs\Design\[ActionPane:ActionPaneHeader]\[ActionPaneTab:General]\[ButtonGroup:Gener

\Design\ActioPaneHeader\General\GeneralSetup\KitShowLinesBtn\Methods



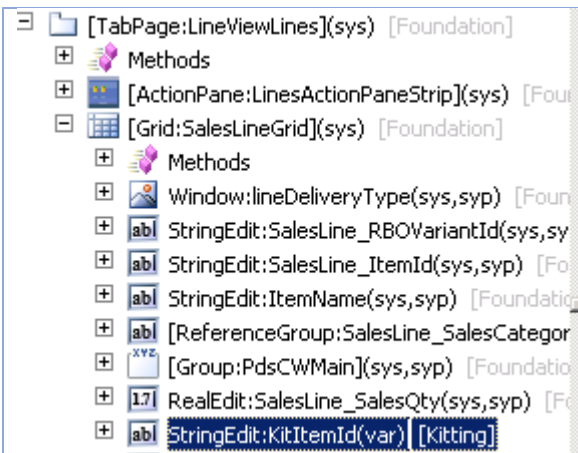
**\Forms\SalesTable\Designs\Design\[[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]
ActionPaneTab:LineOverviewActionTab]\[ButtonGroup:LineOverviewActionButtonGroup]\[MenuButton:LineStripUp**

**\Forms\SalesTable\Designs\Design\[[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]
ActionPaneTab:LineOverviewActionTab]\[ButtonGroup:LineOverviewActionButtonGroup]\[MenuButton:LineStripUp**

**\Design\[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]\[Tab:LineViewTab]\[TabPa
DbClick**

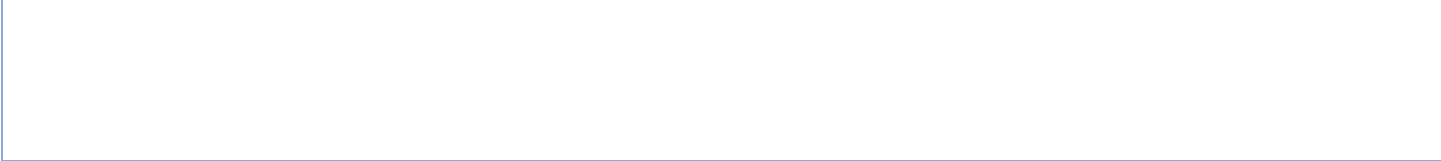
**\Design\[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]\[Tab:LineViewTab]\[TabPa
ontextMenu**

\Design\[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]\[Tab:LineViewTab]\[TabPa



\Design\[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]\[Tab:LineViewTab]\[TabPa

\Design\[Tab:MainTab]\[TabPage:TabPageDetails]\[Tab:DetailsTab]\[TabPage:LineView]\[Tab:LineViewTab]\[TabPa
neDelivery]\[Group:SalesAvailableDlvDatesLineGroup]\MenuItemButton:SalesAvailableDlvDatesLine\Methods\click



Weird... after deleting from var layer, SalesTable still shows in var layer...

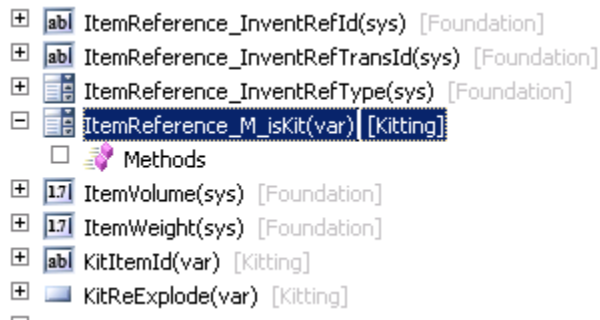
Exported just var layer, found this...

```

1  Exportfile for AOT version 1.0 or later
2  Formatversion: 1
3
4  ***Element: FRM
5
6  ; Microsoft Dynamics AX Forms unloaded
7  ; -----
8  FRMVERSION 12
9
10 FORM #SalesTable
11     PROPERTIES
12         Name                #SalesTable
13         FormTemplate        #DetailsPage
14         InteractionClass    #SalesTableInteraction
15         Origin               #{0B0A0000-0000-1000-B787-EA7473546162}
16     ENDPROPERTIES
17
18     METHODS
19     ENDMETHODS
20     OBJECTBANK
21     ENDOBJECTBANK
22
23     PARTREFERENCES
24     ENDPARTREFERENCES
25
26     DESIGN
27         CONTAINER
28             CONTROL COMBOBOX
29                 PROPERTIES
30                     Name                #ItemReference_M_isKit
31                     AllowEdit          #No
32                     ElementPosition    #1252698793
33                     HierarchyParent    #ItemReference
34                     DataSource         #SalesLine
35                     DataMethod         #isKit
36                 ENDPROPERTIES
37
38                 METHODS
39                 ENDMETHODS
40             ENDCONTROL
41         ENDCONTAINER
42     ENDDESIGN
43
44     PERMISSIONS #Permissions
45         PROPERTIES
46         ENDPROPERTIES
47     ENDPERMISSIONS
48
49     ENDFORM
50
51
52
53
54

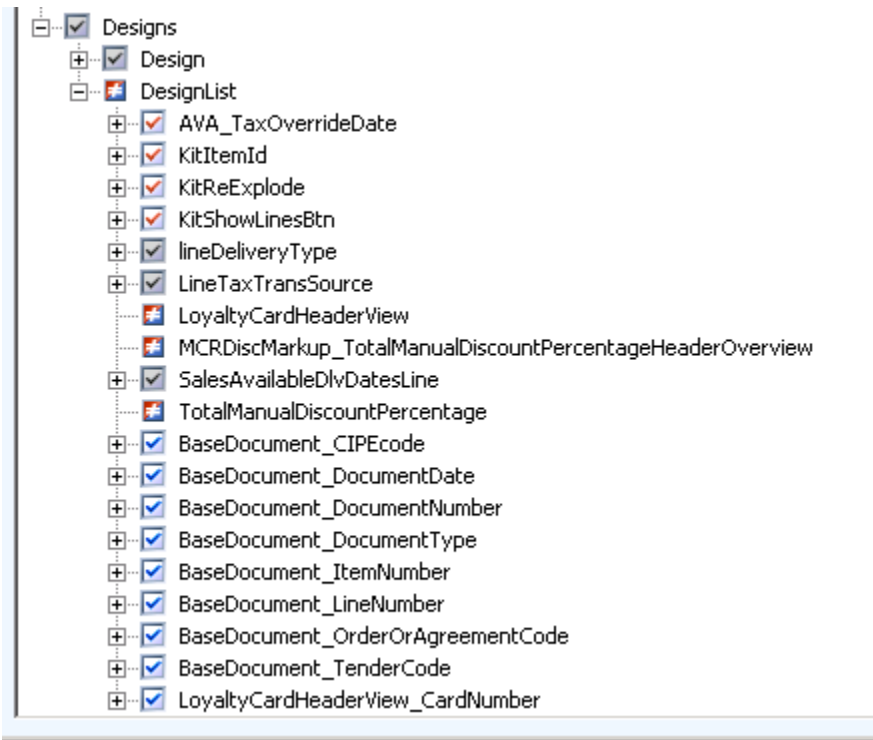
```

Confirmed in GOLD that this control exists.



CountryRegionCodes	
CountryRegionContextField	
DataField	
DataMethod	isKit
DataSource	SalesLine
DisplayLength	Auto
DragDrop	None
Enabled	Yes
EnumType	
ExtendedDataTypes	

But compare process doesn't see it in var... (GOLD)



Find: All nodes

Name & location | Date | Advanced | Filter | Properties

Search: All nodes ▼

Named:

Containing text: isKit

Look in: \Forms\SalesTable

Name	From the source	Path	Size	Run on
ItemReference...	[2] Name #ItemReference_M_isKit	\Forms\SalesTable\Designs\DesignList		
A ItemRefere...	[95] DataMethod #isKit			

Find: All nodes

Name & location | Date | Advanced | Filter | Properties

Search: All nodes ▼

Named:

Containing text: ItemReference_M_isKit

Look in: \Forms\SalesTable

Name	From the source	Path	Size	Run on
ItemReference...	[2] Name #ItemReference_M_isKit	\Forms\SalesTable\Designs\DesignList		

Doesn't seem to be referenced anywhere else, so ignoring.

Worked through the list above to add changes back to var layer for the SalesTable form.

After re-building var layer changes in DEV, one compile error on the Salestable form object remains:

Compiler output

1 Error 0 Warnings 0 Best practices 0 Tasks Reset Edit Setup >>



Description	Path
Unknown source for data method getItemBrand_MX. The method does not exist in the data source or table.	\Forms\SalesTable\Designs\DesignList

Checked GOLD, determined that this method exists in the syp layer. Something else must have promoted the data source above this layer and eliminated this method?

Checked DEV:

Comparison

Name & location | **Advanced**

	\Forms\SalesTable (var)
	\Forms\SalesTable (syp)

☒ SalesTable (var)/SalesTable (syp)

- ☒ Methods
- ☒ Data Sources
 - ☒ SalesTable
 - ☒ SalesLine
 - ☒ Methods
 - ☒ kitItemId
 - ☒ create
 - ☒ executeQuery
 - ☒ init
 - ☒ write
 - ☒ delete
 - ☒ active
 - ☒ displayOption
 - ☒ validateWrite
 - ☒ getItemBrand_MX
 - ☒ Fields
 - ☒ MCRSalesLineDropShipment
 - ☒ RetailSalesTable



```

[ 1] /// <summary>
[ 2] ///     Displays the brand for the item.
[ 3] /// </summary>
[ 4] /// <param name="_salesLine">
[ 5] ///     A buffer of the sales line record selected on
[ 6] /// </param>
[ 7] /// <returns>
[ 8] ///     The brand code.
[ 9] /// </returns>
[10] display BrandCode_MX getItemBrand_MX(SalesLine _sales
[11] {
[12]     return _salesLine.inventTable().BrandCodeId_MX;
[13] }
          
```

Checked DEV ISV layer:

Comparison

Name & location | **Advanced**

	\Forms\SalesTable (isv)
	\Forms\SalesTable (syp)

☒ SalesTable (isv)/SalesTable (syp)

- ☒ Methods
- ☒ Data Sources
 - ☒ SalesTable
 - ☒ SalesLine
 - ☒ Methods
 - ☒ init
 - ☒ write
 - ☒ delete
 - ☒ active
 - ☒ getItemBrand_MX
 - ☒ Fields
 - ☒ MCRSalesLineDropShipment
 - ☒ RetailSalesTable

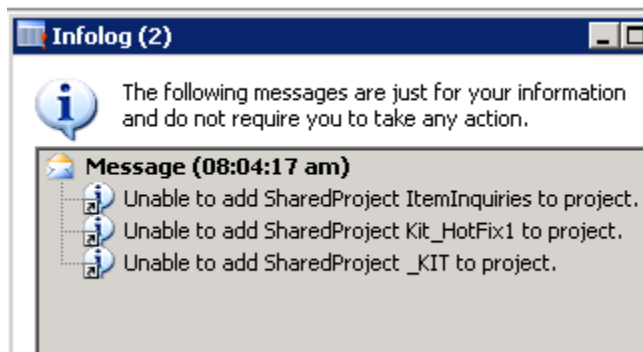
```

[ 1] /// <summary>
[ 2] ///     Displays the brand for the item.
[ 3] /// </summary>
[ 4] /// <param name="_salesLine">
[ 5] ///     A buffer of the sales line record selected on
[ 6] /// </param>
[ 7] /// <returns>
[ 8] ///     The brand code.
[ 9] /// </returns>
[10] display BrandCode_MX getItemBrand_MX(SalesLine _sales:
[11] {
[12]     return _salesLine.inventTable().BrandCodeId_MX;
[13] }
          
```

Added method back in DEV/VAR

Compiled SalesTable form: OK

Created project from Kitting model...



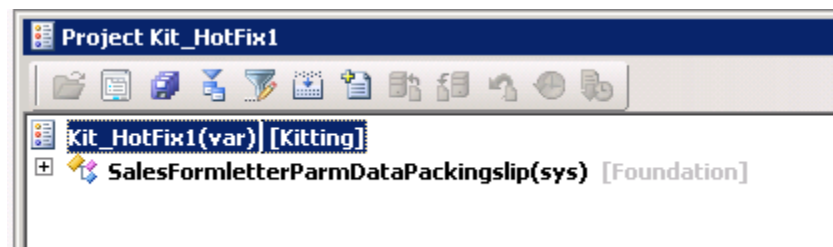
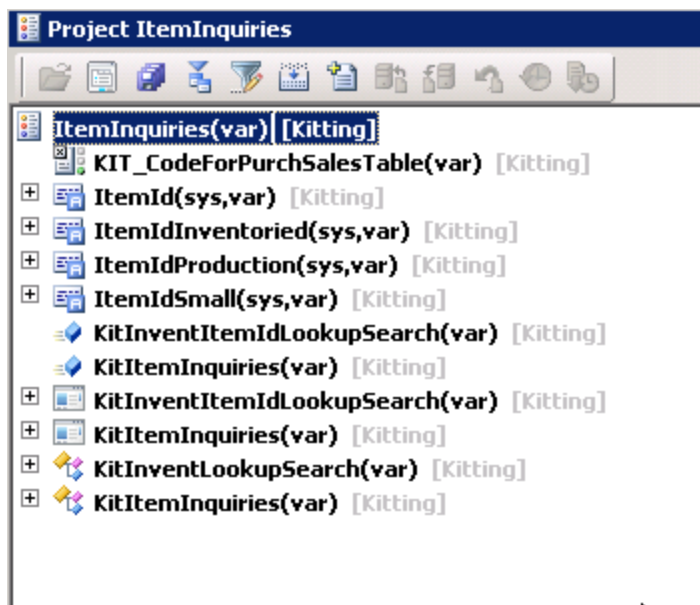
Reviewed and found a missing object: TaxSalesQuotation.

Confirmed code was already migrated, so just need to move object to Kitting model.

Moved \Forms\SalesTable to Kitting model.

Re-checked: OK

Checked other projects:



Switched to cus layer and reviewed Avalara Bug Fixes to make sure they are still there. Compiled the objects in the LHA_AvalaraBugs project: One error:

Compiler output		
	Description	Path
	Variable kitShowLines has not been declared.	\Forms\SalesQuotationT



Checked Gold: same error.

Back in var layer, kitting model

Found the problem:

Comparison

Name & location | Advanced

	\Forms\SalesQuotationTable\Methods\classDeclaration (cus)
	\Forms\SalesQuotationTable\Methods\classDeclaration (var)

classDeclaration (cus)/classDeclarati

[53]	MCROrderEventType	mcrHeaderPriceOverri
[54]	CustAccount	convertedCustAccount.
[54]	CustAccount	mcrConvertedCustAcco
[55]	boolean	mcrCalledFromCampaig
[56]	NoYes	enableDirectedSelling
[57]	NoYes	enableOrderPriceCont
[58]	MCROrderParameters	mcrOrderParameters;
[59]	SalesParameters	salesParameters;
[60]	CustParameters	custParameters;
[61]	// Margin Alerts	
[62]	Map	marginMap;
[63]	// <GBR>	
[64]	SalesPurchOperationTypeController_BR	salesPur
[65]	// </GBR>	
[66]		
[67]	SalesQuotationToLineUpdatePrompt	salesQuotati
[68]		
[69]	#define.Cancel('Cancel')	
[70]	SalesQuotationLine	defaultSalesQuota
[71]	RecVersion	salesQuotationTab
[72]	SysExtensionSerializerFormRunHelper	formRunHelper
[73]	boolean	inDeleteMarked;
[74]		
[75]	boolean	isCalcTax; //
[76]	//>Kit	
[77]	#KIT	
[78]	#if.KitEnabled(true)	
[79]	boolean	kitShowLines;
[80]	QueryBuildDataSource	kitQDS;
[81]	QueryBuildRange	kitQR;
[82]	#endif	
[83]	//<Kit	
[761]	}	

Deleted method from cus layer, promoted to var and re-implemented variable change. Compiles OK now. Checked-in on cus.

Ran compile on entire _KIT project: one error.

Compiler output

Description	Path
 Variable isCalcTax has not been declared.	\Forms\SalesQuotationP

Forms\SalesQuotationProjTable - Editor

classDeclaration
isCalcTax

```

public boolean isCalcTax(boolean _isCalcTax = isCalcTax)
{
    isCalcTax = _isCalcTax;
    return isCalcTax;
}

```

Comparison			
Name & location		Advanced	
[Forms\SalesQuotationProjTable\Methods\classDeclaration (var)			
[Forms\SalesQuotationProjTable\Methods\classDeclaration (isv)			
classDeclaration (var)/classDeclaration (isv)	[19]	LogisticsLocationRecId	postalAddressLoc
	[20]		
	[21]	DimensionDefaultingController	dimensionDefault
	[22]	DimensionDefaultingController	dimensionDefault
	[23]		
	[24]	BankLGFeatureChecker	bankLGFeatureChe
	[25]		
	[26]	ImageListAppl	imageList;
	[27]		
	[28]	boolean	keepName; // Whe
	[29]		
	[30]	// <GIN>	
	[31]	boolean	indiaCountryChec
	[32]	TaxExciseTypeController_IN	taxExciseTypeCon
	[33]	boolean	exciseEnabled;
	[34]	#ISOCountryRegionCodes	
	[35]	// </GIN>	
	[36]	//>Kit	
	[37]	#KIT	
	[38]	#if.KitEnabled(true)	
	[39]	boolean	kitShowLines;
	[40]	QueryBuildDataSource	kitQDS;
	[41]	QueryBuildRange	kitQR;
	[42]	#endif	
	[43]	//<Kit	
	[34]		
	[35]	boolean	isCalcTax; /
	[44]	}	
Read-only properties			
Name		#classDeclaration	

```

26      ImageListAppl          imageUrl;
27
28      boolean                keepName; // When set to false Name is defaulted from categ
29
30      // <GIN>
31      boolean                indiaCountryCheck;
32      TaxExciseTypeController_IN taxExciseTypeController;
33      boolean                exciseEnabled;
34      #ISOCountryRegionCodes
35      // </GIN>
36  //>Kit
37  #KIT
38  #if.KitEnabled(true)
39      boolean                kitShowLines;
40      QueryBuildDataSource    kitQDS;
41      QueryBuildRange        kitQR;
42  #endif
43  //<Kit
44  }
45  |

```

```

ImageListAppl          imageUrl;

//>Kit
#KIT
#if.KitEnabled(true)
    boolean                kitShowLines;
    QueryBuildDataSource    kitQDS;
    QueryBuildRange        kitQR;
#endif
//<Kit





```

Deleted from var layer

Added code back. Now compiles OK.

Re-tried compiling entire project: OK.

Some BP's and orphaned code needs cleaned up:

Compiler output		
<div> <div>0 Errors</div> <div>0 Warnings</div> <div>13 Best practices</div> <div>0 Tasks</div> <div>Reset</div> <div>Edit</div> <div>Setup</div> <div>>></div> </div>		
	Description	Path
	Field is not a member of a field group. Consider setting the Visible property to No, or build the user interface using field groups.	\Data Dictionary\Tables
	Field is not a member of a field group. Consider setting the Visible property to No, or build the user interface using field groups.	\Data Dictionary\Tables
	Fields using RefRecId or a derived type must have a relation defined for that field.	\Data Dictionary\Tables
	Field is not a member of a field group. Consider setting the Visible property to No, or build the user interface using field groups.	\Data Dictionary\Tables

<https://docs.microsoft.com/en-us/dynamicsax-2012/appuser-itpro/how-to-export-and-import-a-model>

Per the above document, "In general, never delete a model and then reinstall it.", so I won't uninstall it from GOLD after all. If there are any compile errors in GOLD after importing the updated axmodel file, I'll deal with them in GOLD.

Opened ax shell

Ran command:

Export-AXModel -Model KIT -File Kitting_LHA.axmodel -Config LHA_2012R3_DEV

```
PS C:\temp> Export-AXModel -Model KIT -File Kitting_LHA.axmodel -Config LHA_2012R3_DEV
Model file C:\Windows\system32\Kitting_LHA.axmodel successfully exported from model 'KIT (CodePlex)'.
PS C:\temp>
```

Moved file to X:\Reference\Dynamics AX\Customizations\R3\DAXKitting

Copied to C:\Temp

Ran AXBuild for GOLD environment first to make sure it compiles OK before making changes.

```
C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>axbuild xpp
compileall /s=03

Microsoft (R) Dynamics AX Build Utility 6.3.164.0
Copyright (C) Microsoft Corporation. All rights reserved.

*** Compile all starting at 6/2/2020 9:13:18 AM.
*** Kicking off header compilation (pass 1) with 24 workers.
*** Kicking off full compilation (pass 2) with 24 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Kicking off full compilation (pass 3) with 24 workers.
*** Kicking off full compilation (pass 4) with 1 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Compile all completed at 6/2/2020 9:31:14 AM. Total duration was 00:17:55.5275420.

Log files are located at 'C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\Log'
C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>
```

Installed model in GOLD:

Install-AXModel -File C:\Temp\Kitting_LHA.axmodel -Config LHA_2012R3_Gold

```

PS C:\temp> Install-AXModel -File C:\Temp\Kitting_LHA.axmodel -Config LHA_2012R3_Gold

The model is not signed. Are you sure you want to install this model (Y/N)?
Continue with this operation?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y

Are you sure you want to overwrite this or these model(s) (Y/N)?
Continue with this operation?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): y
The model file C:\Temp\Kitting_LHA.axmodel was successfully imported into the U
ar layer.
Imported model: KIT (CodePlex).
\Classes: 42
\Data Dictionary\Base Enums: 3
\Data Dictionary\Extended Data Types: 11
\Data Dictionary\Tables: 12
\Forms: 10
\Jobs: 1
\Label Files: 1
\Macros: 1
\Menu items\Display: 3
\Menus: 2
\Projects\Private: 2
\Projects\Shared: 3
\Security\Duties: 1
\Security\Privileges: 1
\Security\Process Cycles: 1
\Security\Roles: 1

Because of the changes made to the model store, it is highly recommended to
restart the AOS, run appropriate scripts, and compile and synchronize the
application. Not doing that may cause runtime errors and loss of data.
PS C:\temp>

```

Restarted Gold

Ran axbuild to compile Gold.

Compiled OK:

```

C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>axbuild xpp
compileall /s=03

Microsoft (R) Dynamics AX Build Utility 6.3.164.0
Copyright (C) Microsoft Corporation. All rights reserved.

*** Compile all starting at 6/2/2020 9:45:41 AM.
*** Kicking off header compilation (pass 1) with 24 workers.
*** Kicking off full compilation (pass 2) with 24 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Kicking off full compilation (pass 3) with 24 workers.
*** Kicking off full compilation (pass 4) with 1 workers.
*** Kicking off full compilation (pass 3) with 1 workers.
*** Compile all completed at 6/2/2020 10:02:31 AM. Total duration was 00:16:49.2
779177.

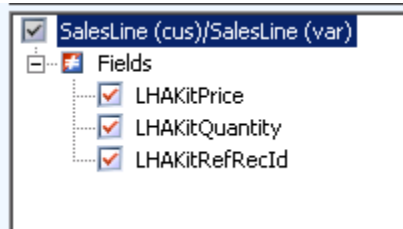
Log files are located at 'C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2
012R3_Gold\Log'

C:\Program Files\Microsoft Dynamics AX\60\Server\LHA_2012R3_Gold\bin>

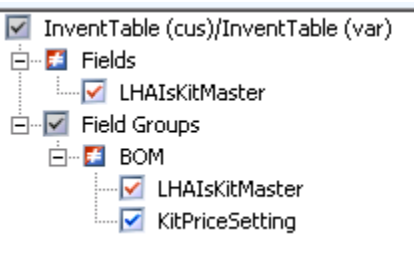
```

Rolled back "LHA_TypeIIKits project:

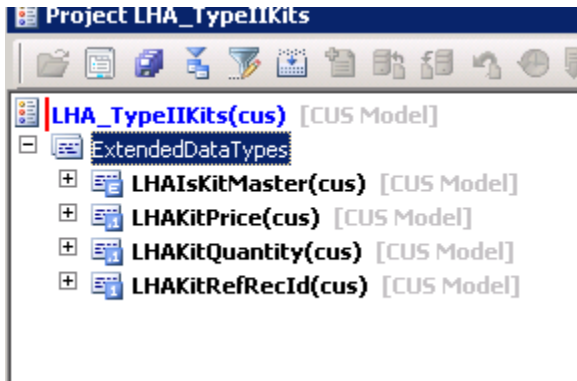
- Deleted \Tables\SalesLine from cus layer



- Deleted \Tables\InventTable from cus layer



- Deleted EDTs:



Checked-in changes.








Check in

Model elements

Work items

Description

LHA_TypeIIKits: Rolled back in favor of DAXKitting

Include		Path	Action
<input checked="" type="checkbox"/>		\Projects\Shared\LHA_TypeIIKits	edit
<input checked="" type="checkbox"/>		\Data Dictionary\Tables\SalesLine	delete
<input checked="" type="checkbox"/>		\Data Dictionary\Tables\InventTable	delete
<input checked="" type="checkbox"/>		\Data Dictionary\Extended Data Types\LHAKitRefRecId	delete
<input checked="" type="checkbox"/>		\Data Dictionary\Extended Data Types\LHAKitQuantity	delete
<input checked="" type="checkbox"/>		\Data Dictionary\Extended Data Types\LHAKitPrice	delete
<input checked="" type="checkbox"/>		\Data Dictionary\Extended Data Types\LHAKitMaster	delete

Deleted project and checked-in.

Opened GOLD cus layer, ran sync

Info get Number of failures: 0

Number of Warnings: 0

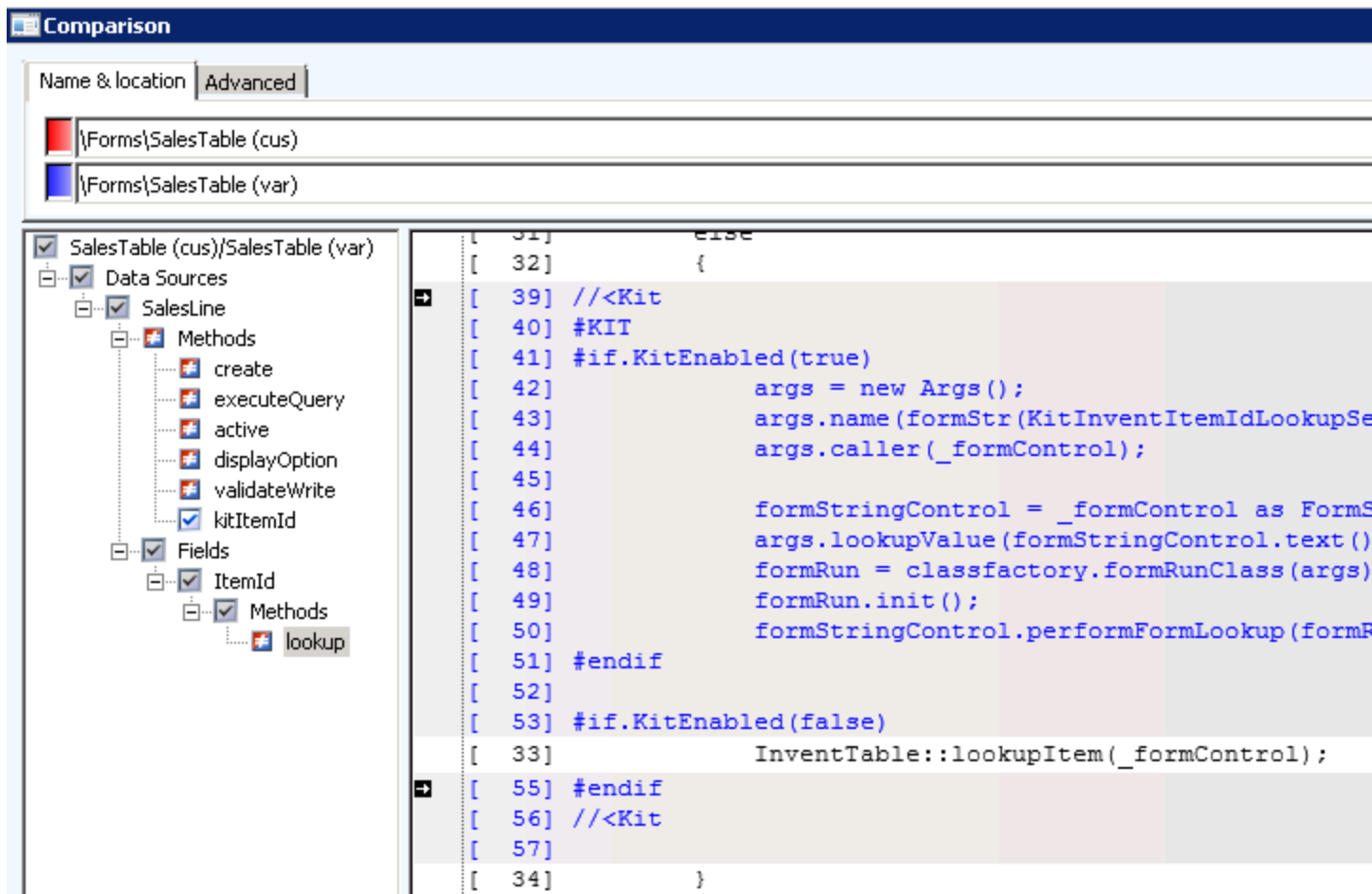
Have resolvable Warnings: False

Number of Operations: 0

Number of Conflicts: 0

Objects are no longer showing as under version control. Deleted in GOLD cus.

Except Gold has something in \Forms\SalesTable on cus layer? All were related to DAXKitting



Deleted from cus.



Started build script.

```
buildno.txt copied.
New build completed successfully.
Drop folder: C:\builds\6.3.0.29
-----
Script completed succesfully.
Press any key to continue . . .
```

Started import latest script

```
in Reports deployed.
Saving new build number ...
Build number saved.
New build completed successfully.
-----
Script completed succesfully.
Press any key to continue . . .
```

Configured Kitting Formletter Setup in GOLD:

Kitting form letter setup (1 - lha) - New Record								
File  New  Delete								
Table source	Account code	Account selection	Valid for	Item relation	Quotation	Project quotation	Quotation confirmation	Confirmation
Customer t...	All		All		Kit+lines	Kit+lines	Kit+lines	Kit+lines

And TEST.

Setup 00006W5

00006W5 : Wick for Gimbaled Berth Lamp P

Product type: Item Product subtype: Product

Product model group:

▶ Purchase

▶ Promote

▶ Deliver

▶ Sell

▶ Foreign trade

▶ Manage inventory

▲ Engineer

BOM unit:

Constant scrap:

Variable scrap:

Costing level:

Planning level:

Phantom: ☒

Auto-report as finished: ☐

Measurement

Kit pricing field is missing.

Engineer		Production	
BOM unit:		Production pool:	
Constant scrap:		Production group:	
Variable scrap:		Property:	
Costing level:	0	Arrival:	<input type="checkbox"/>
Planning level:	0	Flushing principle:	Start
Phantom:	<input checked="" type="checkbox"/>	Product Builder	
Auto-report as finished:	<input type="checkbox"/>	Modeling enabled:	<input type="checkbox"/>
Measurement		Hide dialog box:	<input type="checkbox"/>
Height:		Hide approval dialog:	<input type="checkbox"/>
Width:		Autostart configuration:	<input type="checkbox"/>
Depth:		Mandatory configuration:	<input type="checkbox"/>
Density:		Formula planning	
Calculation		Production type:	BOM
Calculation group:	DEF	Planning formula:	

Training materials show this...

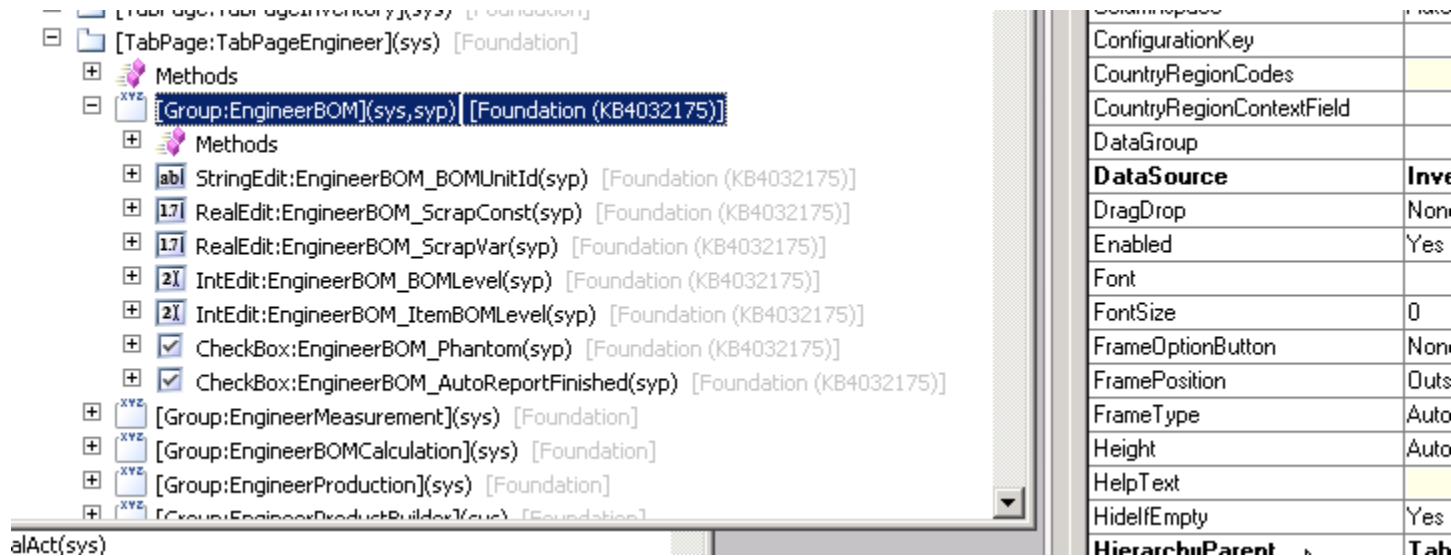
Bill of materials		Flush
BOM unit:	ea	Prod
Constant scrap:		Modi
Variable scrap:		Hide
Level:	0	Hide
Phantom:	<input checked="" type="checkbox"/>	Auto:
Kit price setting:	<div> <div>um up prices</div> <div>Use kit prices</div> <div>Sum up prices</div> </div>	Mane
Calculation		Form
Calculation group:		Prodi
		Planir

KitPriceSetting field is showing in the BOM field group on the InventTable table.

- BOM(sys, var) [Kitting]
 - BOMUnitId
 - ScrapConst
 - ScrapVar
 - BOMLevel
 - AutoReportFinished
 - Phantom
 - KitPriceSetting
- BOMCalculation(sys) [Foundation]
 -

EcoResProductDetailsExtended form is not in the project though.

And the EngineerBOM group control doesn't use the table field group....



Logged-in to DEV/var

Added EcoResProductDetailsExtended to project

Added KitPriceSetting to EngineerBOM group, moved to bottom.

Moved EcoResProductDetailsExtended to Kitting model

Exported model

```
PS C:\temp> Export-AXModel -Model KIT -File Kitting_LHA.axmodel -Config LHA_2012
R3_DEU
Model file C:\Windows\system32\Kitting_LHA.axmodel successfully exported from m
odel 'KIT <CodePlex>'.
PS C:\temp>
```

Moved to temp

Copied to X:\Reference\Dynamics AX\Customizations\R3\DAXKitting

Installed model in GOLD.

```
PS C:\temp> Install-AXModel -File C:\Temp\Kitting_LHA.axmodel -Config LHA_2012R3_Gold

The model is not signed. Are you sure you want to install this model (Y/N)?
Continue with this operation?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y

Are you sure you want to overwrite this or these model(s) (Y/N)?
Continue with this operation?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): Y
The model file C:\Temp\Kitting_LHA.axmodel was successfully imported into the U
ar layer.
Imported model: KIT (CodePlex).
\Classes: 42
\Data Dictionary\Base Enums: 3
\Data Dictionary\Extended Data Types: 11
\Data Dictionary\Tables: 12
\Forms: 11
\Jobs: 1
\Label Files: 1
\Macros: 1
\Menu items\Display: 3
\Menus: 2
\Projects\Private: 2
\Projects\Shared: 3
\Security\Duties: 1
\Security\Privileges: 1
\Security\Process Cycles: 1
\Security\Roles: 1

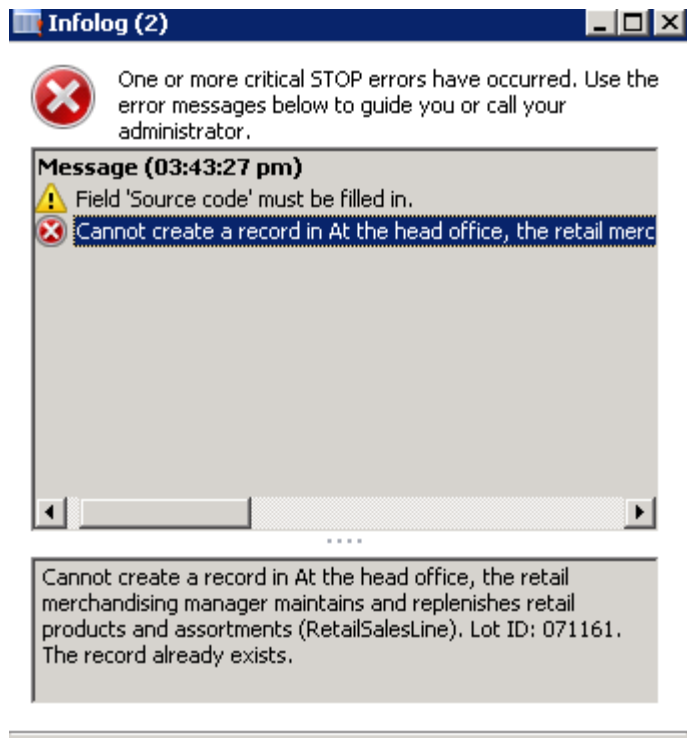
Because of the changes made to the model store, it is highly recommended to
restart the AOS, run appropriate scripts, and compile and synchronize the
application. Not doing that may cause runtime errors and loss of data.
PS C:\temp>
```

Opened GOLD client, skipped checklist.

Confirmed the update is in the form.

Ran build script

Got error when attempting to add kit in call center order:



Warning Message (03:43:27 pm) Field 'Source code' must be filled in.

Error Message (03:43:27 pm) Cannot create a record in At the head office, the retail merchandising manager maintains and replenishes retail products and assortments (RetailSalesLine). Lot ID: 071161.

The record already exists.

Error "Field 'Source code' must be filled in with a valid source code." happens here:

Microsoft Dynamics AX Debugger [break] - [\Forms\SalesTable\Data Sources\SalesLine\Methods\validateWrite]

File Edit View Debug Window Help

```
12     salesLine.DeliveryDateControlType = SalesDeliveryDateControlType::None;
13 }
14
15 if (!salesLine.isAgreementLinkValid(true))
16 {
17     ret = salesLine.unlinkAgreementLinePrompt(0, tableNum(SalesLine), false);
18 }
19
20 this.updateInventDim();
21
22 salesTable_ds.packExtensions();
23 this.packExtensions();
24 ret = super() && ret;
25 this.unpackExtensions();
26 salesTable_ds.unpackExtensions();
27
28 if (!askingModifyInvoiced && ret)
```

Location	Path	Line
	\Forms\SalesTable\Data Sources\SalesLine\Methods\validateWrite	26
	\Classes\FormDataSource\leaveRecord	0
	\Classes\FormDataSource\Next	0
	\Classes\FormRun\Task	0
	\Forms\SalesTable\Methods\Task	12
	\Classes\SysSetupFormRun\Task	27

Error is thrown here:

Microsoft Dynamics AX Debugger [break] - [\Forms\SalesTable\Data Sources\SalesLine\Methods\write]

```

74
75     // Want to know if inserting because will show form for gift certificates
76     // if inserting but not if modifying.
77     if (salesLine.RecId == 0)
78     {
79         onInsert = true;
80     }
81     // <GEEPL>
82     serviceTariffNumber = taxServiceTariffLine.ServiceTariffNumber;
83
84     // </GEEPL>
85
86     this.packExtensions();
87     super();
88     this.unpackExtensions();
89
90     // To keep all information extra explicit and direct follow the newSalesLine

```

Field 'Source code' must be filled in.

Cannot create a record in At the head office, the retail merchandising manager maintains and replenishes retail products and assortments (RetailSalesLine). Lot ID: 071166.











The record already exists.

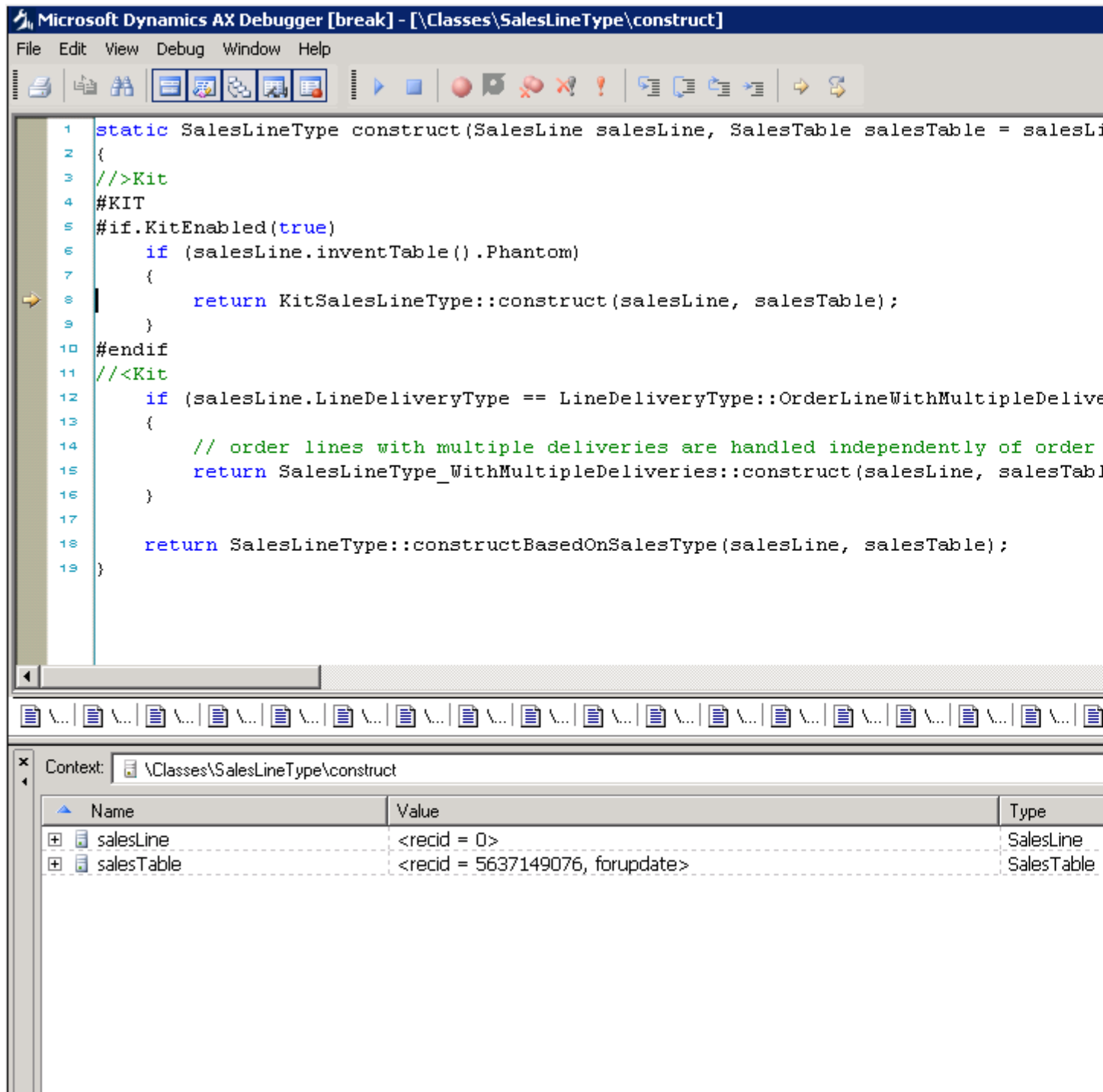
Step into...

Location	Path	Line
	\Classes\Application\ ttsNotifyBegin	1
	\Classes\xRecord\ddbOpInTransaction	0
	\Classes\FormDataSource\write	0
	\Forms\SalesTable\Data Sources\SalesLine\Methods\write	87
	\Classes\FormDataSource\leaveRecord	0
	\Classes\FormDataSource\Next	0
	\Classes\FormRun\task	0
	\Forms\SalesTable\Methods\task	12
	\Classes\SysSetupFormRun\task	27

Something is happening, since before moving off of the field the source code is there...

Stepped through with F11.

Location	Path	Line
	\Data Dictionary\Tables\SalesLine\Methods\setInventSumLogTTS	17
	\Data Dictionary\Tables\SalesLine\Methods\Insert	39
	\Classes\wRecord\dbOpInTransaction	0
	\Classes\FormDataSource\write	0
	\Forms\SalesTable\Data Sources\SalesLine\Methods\write	87
	\Classes\FormDataSource\leaveRecord	0
	\Classes\FormDataSource\Next	0
	\Classes\FormRun\task	0
	\Forms\SalesTable\Methods\task	12
	\Classes\SysSetupFormRun\task	27

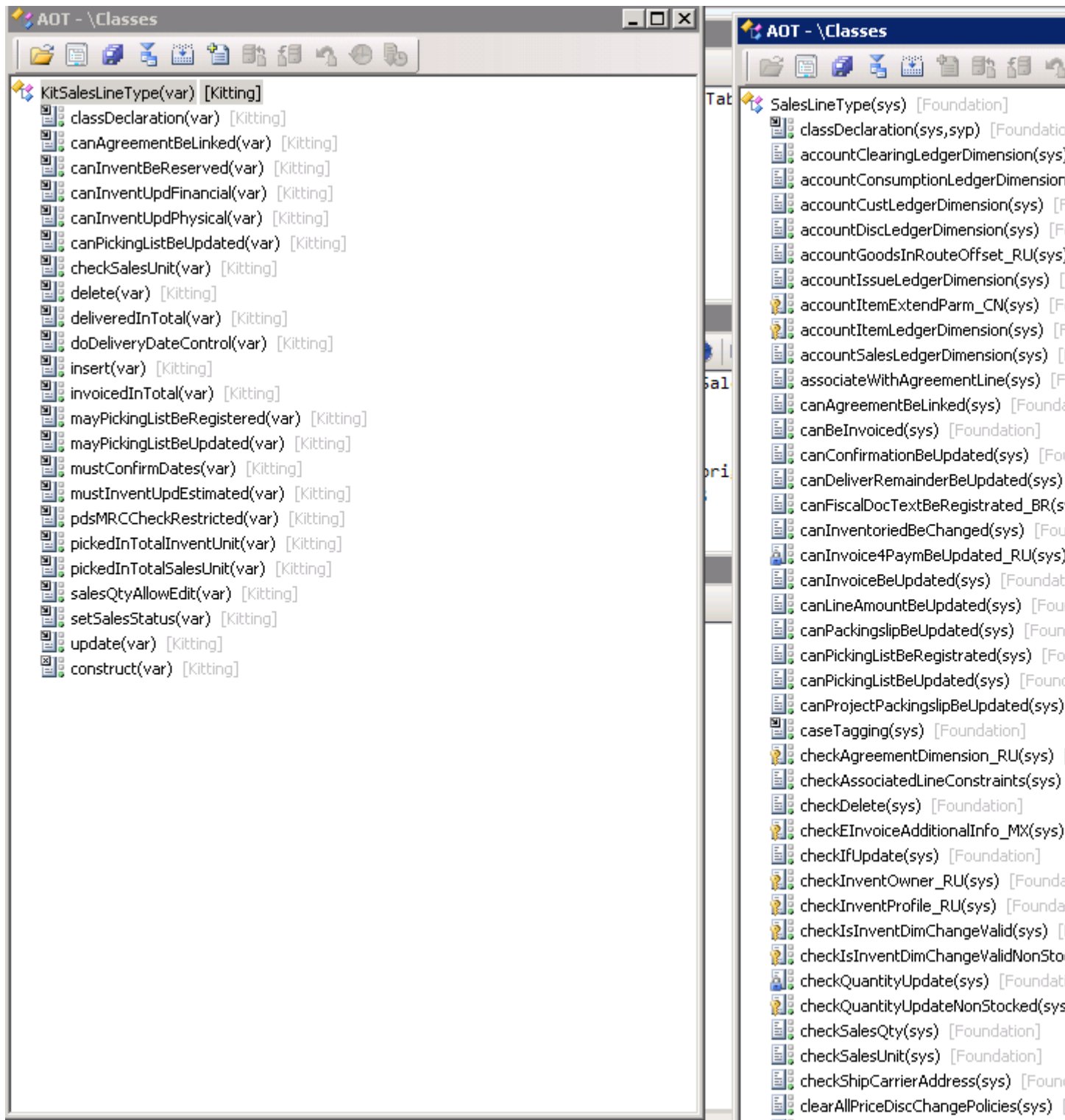


The above might be the problem...

Tables\SalesLine\Methods\Insert line 45 is where this warning appears:
Field 'Source code' must be filled in.

Line 66 throws the error.

KitSalesLineType extends SalesLineType



Why does it even have the insert method, since it just calls super?

The screenshot shows the 'KitSalesLineType - Editor' window. The left sidebar has a tree view with 'classDeclaration', 'construct', and 'insert' items. The 'insert' item is selected. The main editor area displays the following code:

```

public void insert(boolean dropInvent = false,
                  boolean findMarkup = true,
                  Common childBuffer = null,
                  boolean _skipCreditLimitCheck = false)
{
    super(dropInvent, findMarkup, childBuffer, _skipCreditLimitCheck);
}

```

These lines here are suspect...

The screenshot shows the 'SalesLineType - Editor' window. The left sidebar has a tree view with 'classDeclaration', 'construct', 'constructBasedOnSalesType', 'new', and 'insert' items. The 'insert' item is selected. The main editor area displays code lines 156 to 169. Lines 164 and 165 are highlighted in blue:

```

156 kitSalesLine = KitSalesLine::find(salesLine.InventTransId,true)
157 kitSalesLine.KitInventTransId = salesLine.InventTrans
158 kitSalesLine.KitExplosion = KitExplosion::Yes;
159 kitSalesLine.KitExplodedQty = salesLine.SalesQty;
160 salesLine.DeliveryDateControlType = SalesDeliveryDateCont
161 if (!kitSalesLine)
162     kitSalesLine.KitPriceSetting = salesLine.inventT
163 kitSalesLine.write();
164 kitExplode_SalesLine = KitExplode_SalesLine::construct(salesLin
165 kitExplode_SalesLine.insertKit();
166 salesLine.reread();
167 }
168 #endif
169 //<Kit

```

The kit lines might be missing the source code ID and the retailsalesline record.

Here is where the mcrSalesLine record is created from the salesLine.

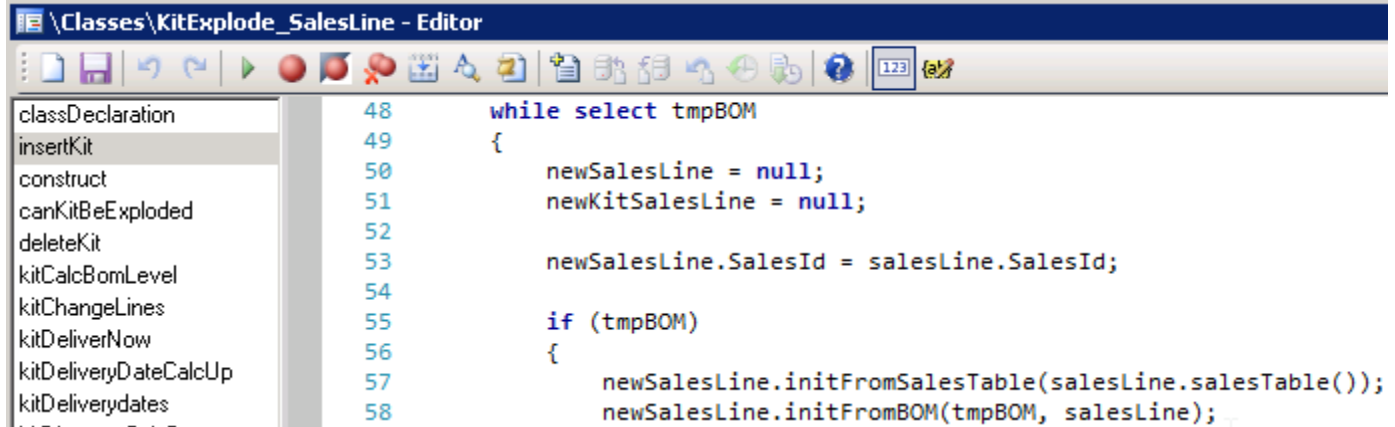
The screenshot shows the 'SalesLineType - Editor' window. The left sidebar has a tree view with 'classDeclaration', 'construct', 'constructBasedOnSalesType', 'new', and 'insert' items. The 'insert' item is selected. The main editor area displays code lines 179 to 195. Lines 189 and 190 are highlighted in blue:

```

179 // As the Salesline_IN is used later before its form data s
180 // DB, insert it at here to make it accessible from DB.s
181 salesline_IN = SysExtensionSerializerFormRunHelper::getExte
182 if (salesline_IN.RecId == 0 && salesline_IN.isFormDataSource
183 {
184     salesline_IN.SysExtensionSerializerExtensionMap::insert
185 }
186 }
187 // </GIN>
188
189 mcrSalesLine = salesline.mcrSalesLine();
190 if (mcrSalesLine.RecId == 0 && mcrSalesLine.isFormDataSource())
191 {
192     mcrSalesLine.SysExtensionSerializerExtensionMap::insertAfte
193 }
194
195 mcrSalesLineDropShipment = salesline.mcrSalesLineDropShipment()

```

Here is where it loops through the BOM:

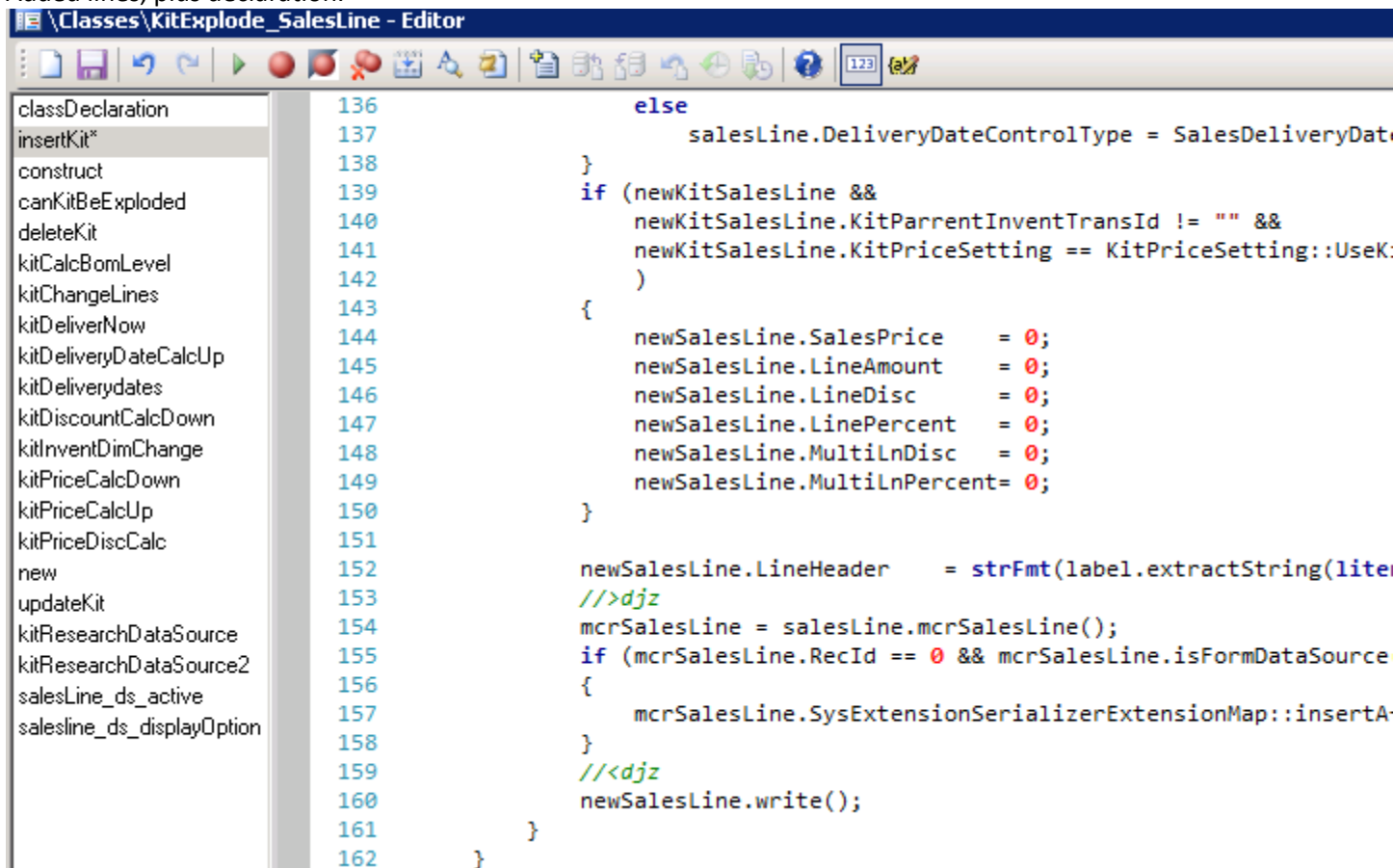


```

classDeclaration      48      while select tmpBOM
insertKit              49      {
                        50          newSalesLine = null;
                        51          newKitSalesLine = null;
                        52
                        53          newSalesLine.SalesId = salesLine.SalesId;
                        54
                        55          if (tmpBOM)
                        56          {
                        57              newSalesLine.initFromSalesTable(salesLine.salesTable());
                        58              newSalesLine.initFromBOM(tmpBOM, salesLine);

```

Added lines, plus declaration.



```

classDeclaration      136
insertKit*            137      else
                        138          salesLine.DeliveryDateControlType = SalesDeliveryDateControlType;
                        139      }
                        140      if (newKitSalesLine &&
                        141          newKitSalesLine.KitParrentInventTransId != "" &&
                        142          newKitSalesLine.KitPriceSetting == KitPriceSetting::UseKitPriceSetting)
                        143      {
                        144          newSalesLine.SalesPrice      = 0;
                        145          newSalesLine.LineAmount      = 0;
                        146          newSalesLine.LineDisc        = 0;
                        147          newSalesLine.LinePercent     = 0;
                        148          newSalesLine.MultiLnDisc     = 0;
                        149          newSalesLine.MultiLnPercent = 0;
                        150      }
                        151
                        152      newSalesLine.LineHeader = strFmt(label.extractString(literal));
                        153      //>djz
                        154      mcrSalesLine = salesLine.mcrSalesLine();
                        155      if (mcrSalesLine.RecId == 0 && mcrSalesLine.isFormDataSource)
                        156      {
                        157          mcrSalesLine.SysExtensionSerializerExtensionMap::insertA
                        158      }
                        159      //<djz
                        160      newSalesLine.write();
                        161      }
                        162  }

```

Compiled OK. Ran incremental CIL.
Tested: No change.

Find: Methods

Name & location | Date | Advanced | Filter


Search:

Named:

Containing text: Show source code: ☒

Look in: Use selection: ☒

Find now
Stop
New search



Name	From the source	Path	Size

Number of nodes found: 0 - Time consumed: 0:00:00

Where is that table referred to?

Remarked out changes for now.

Added override to RetailSalesLine.insert().

\Data Dictionary\Tables\RetailSalesLine - Editor

insert

```

1 public void insert()
2 {
3     super();
4 }

```

Location	Path	Line
	\Data Dictionary\Tables\RetailSalesLine\Methods\insert	3
	\Data Dictionary\Tables\SalesLine\Methods\writeRetailSalesLine	26
	\Data Dictionary\Tables\SalesLine\Methods\insert	66
	\Data Dictionary\Tables\SalesLine\Methods\createLine	155
	\Classes\KitExplode_SalesLine\insertKit	130
	\Classes\SalesLineType\insert	165
	\Classes\KitSalesLineType\insert	6
	\Data Dictionary\Tables\SalesLine\Methods\Insert	45
	\Classes\wRecord\dbOpInTransaction	0
	\Classes\FormDataSource\write	0
	\Forms\SalesTable\Data Sources\SalesLine\Methods\write	87
	\Classes\FormDataSource\leaveRecord	0
	\Classes\FormDataSource\Next	0
	\Classes\FormRun\Task	0
	\Forms\SalesTable\Methods\Task	12

First time hit: No error recorded.

Second time...

Location	Path	Line
	\Data Dictionary\Tables\RetailSalesLine\Methods\insert	3
	\Data Dictionary\Tables\SalesLine\Methods\writeRetailSalesLine	26
	\Data Dictionary\Tables\SalesLine\Methods\update	38
	\Classes\KitExplode_SalesLine\insertKit	173
	\Classes\SalesLineType\insert	165
	\Classes\KitSalesLineType\insert	6
	\Data Dictionary\Tables\SalesLine\Methods\Insert	45
	\Classes\wRecord\dbOpInTransaction	0
	\Classes\FormDataSource\write	0
	\Forms\SalesTable\Data Sources\SalesLine\Methods\write	87
	\Classes\FormDataSource\leaveRecord	0
	\Classes\FormDataSource\Next	0
	\Classes\FormRun\Task	0
	\Forms\SalesTable\Methods\Task	12
	\Classes\SysSetupFormRun\Task	27

Looks like it should be working for the kit item?


```
25
26     retailSalesLineLocal.insert();
27 }
28 }
29 }
```

Context: \Data Dictionary\Tables\SalesLine\Methods\writeRetailSalesLine

Name	Value	Type
_forceInsert	false	boolean
mcrSourceCodeSetupL...	<recid = 5637145375>	MCRSourceCodeS...
retailSalesLineLocal	<recid = 0>	RetailSalesLine
sourceId	"(NONE)"	str 20
this	<recid = 5637150591>	SalesLine

Variables: Locals | Globals | This

Path: \Data Dictionary\Tables\RetailSalesLine\Methods\insert

This might be one of the problems... it is hitting this method again for the same salesline.

Context: \Data Dictionary\Tables\SalesLine\Methods\writeRetailSalesLine

Name	Value	Type
InvoiceGTDId_RU	""	str30
ItemBOMId	""	str20
ItemId	"00006W5"	str20
ItemPBAId	""	str20
ItemReplaced	No	NoYes
ItemRouteId	""	str20
ItemTagging	No	NoYes
LedgerDimension	0	int64
LineAmount	21.95	real
LineDeliveryType	OrderLine	LineDeliveryType
LineDisc	0	real
LineHeader	""	str80
LineNum	1	real

Variables: Locals | Globals | This

Maybe not a problem... it is updating the master sales line... but what happened with the kit components? Maybe that was the first time?

Microsoft Dynamics AX Debugger [break] - [\Classes\KitExplode_SalesLine\insertKit]

```

157         //mcrSalesLine.SysExtensionSerializerExtensionMap::insertAfterBase
158         //}
159         //<djz
160         newSalesLine.write();
161     }
162 }
163
164 this.kitPriceCalcDown();
165 this.kitPriceDiscCalc();
166 this.kitCalcBomLevel(kitSalesLine, kitSalesLine.KitBomLevel+1);
167
168 this.kitDeliveryDateCalcUp(true); //calculate delivery dates for kit parents
169 salesLine.reread();
170 InventMovement::bufferSetTransQtyUnit(salesLine);
171
172 salesLine.setInventDeliverNow();
173 salesLine.update();
174 InterCompanyUpdateStatus::synchronize(salesLine);
175 ttscommit;
176
177 return ret;
178 }

```

Yeah, the first time was on line 130, which is up in the loop...

```

125         newKitSalesLine.KitPriceSetting           = kitSalesLine.Kit
126
127         newKitSalesLine.write();
128     }
129
130     newSalesLine.createLine(true,false,false,false,true,true,false,true, '
131     newSalesLine.DefaultDimension      = salesLine.DefaultDimension;
132     if (newSalesLine.DeliveryDateControlType != SalesDeliveryDateControlTy
133     {
134         if (!salesLine.inventTable().Phantom)
135             SalesCalcAvailableDlvDates::initDateFields(salesLine);
136     else
137         newSalesLine.DeliveryDateControlType = SalesDeliveryDateControlType

```

Insert on RetailSalesLine is called a third time, and sometime by then the source code problem appears...

Location	Path	Line
	\Data Dictionary\Tables\RetailSalesLine\Methods\insert	3
	\Data Dictionary\Tables\SalesLine\Methods\writeRetailSalesLine	26
	\Data Dictionary\Tables\SalesLine\Methods\Insert	66
	\Classes\wRecord\wbOpInTransaction	0
	\Classes\FormDataSource\write	0
	\Forms\SalesTable\Data Sources\SalesLine\Methods\write	87
	\Classes\FormDataSource\leaveRecord	0
	\Classes\FormDataSource\Next	0
	\Classes\FormRun\task	0
	\Forms\SalesTable\Methods\task	12
	\Classes\SysSetupFormRun\task	27

Call Stack

Field 'Source code' must be filled in.

Remarked out these lines just to see what happens...

```

\Classes\KitExplode_SalesLine - Editor
classDeclaration      161      }
insertKit             162      }
construct             163
canKitBeExploded      164      this.kitPriceCalcDown();
deleteKit             165      this.kitPriceDiscCalc();
kitCalcBomLevel        166      this.kitCalcBomLevel(kitSalesLine, kitSalesLine.KitBomLevel+1);
kitChangeLines        167
kitDeliverNow          168      this.kitDeliveryDateCalcUp(true); //calculate delivery dates for kit
kitDeliveryDateCalcUp  169      //salesLine.reread();
kitDeliverydates       170      //InventMovement::bufferSetTransQtyUnit(salesLine);
kitDiscountCalcDown   171      //
kitInventDimChange     172      //salesLine.setInventDeliverNow();
kitPriceCalcDown       173      //salesLine.update();
kitPriceCalcUp         174      //InterCompanyUpdateStatus::synchronize(salesLine);
kitPriceDiscCalc       175      ttscommit;
new                   176
updateKit             177      return ret;
kitResearchDataSource  178      }

```

I can now add the kit line, no errors, but no kit components either..?

Sales order lines

Add line Add lines Add products Remove Sales order line ▼ Financials ▼ Inventory ▼						
<input type="checkbox"/>	Type	Source code	Variant number	Item number	Product name	Sales category
		(NONE)		00006W5	Wick for Gimbaled Berth Lamp...	
		(NONE)		<input type="text"/>		

Clicked "Show Kits" button, and now it shows the kit components...

Sales order lines

Add line Add lines Add products Remove Sales order line ▼ Financials ▼ Inventory ▼						
<input type="checkbox"/>	Type	Source code	Variant number	Product name	Sales category	Quantity
		(NONE)		Wick for Gimbaled Berth Lamp...		1.0
		(NONE)		Wick for Gimbaled Berth Lamp		5.0

So we are getting close.

Reduced remarking to this:

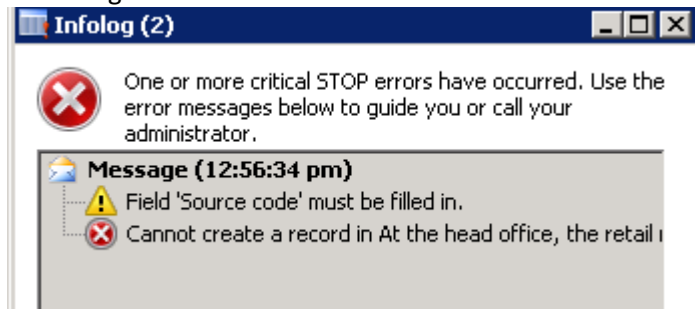
```

167
168     this.kitDeliveryDateCalcUp(true); //calculate delivery dates for kit parents
169     //salesLine.reread();
170     InventMovement::bufferSetTransQtyUnit(salesLine);
171
172     salesLine.setInventDeliverNow();
173     salesLine.update();
174     InterCompanyUpdateStatus::synchronize(salesLine);
175     ttscommit;
176
177     return ret;
178 }

```

Ran incremental CIL.

Tested again: Same error.



Remarked just this line:

Classes\KitExplode_SalesLine - Editor

```

classDeclaration
insertKit*
156
157
158
159
160     newSalesLine.write();
161 }
162 }
163
164     this.kitPriceCalcDown();
165     this.kitPriceDiscCalc();
166     this.kitCalcBomLevel(kitSalesLine, kitSalesLine.KitBomLevel+1);
167
168     this.kitDeliveryDateCalcUp(true); //calculate delivery dates for kit parent
169     salesLine.reread();
170     InventMovement::bufferSetTransQtyUnit(salesLine);
171
172     salesLine.setInventDeliverNow();
173     //salesLine.update();
174     InterCompanyUpdateStatus::synchronize(salesLine);
175     ttscommit;
176
177     return ret;
178 }

```

Generated incremental CIL.

That seems to work so far?

Sales order (2 - lha) - Sales order: 30170036, Daniel Zook

File Sales order Sell Manage Pick and pack Invoice Retail General Warehouse management

Hide kit lines Print management Set up Summary Trade agreements Customer Activities Cases Contact details Customer service Send catalog Supplementary sales items Maintain Reference

30170036 : 00739406 - Daniel Zook

► Sales order header

Sales order lines

Add line Add lines Add products Remove Sales order line Financials Inventory

Type	Source code	Variant number	Product name	Sales category	Quantity	Item number	Unit
	(NONE)		Wick for Gimbaled Berth Lamp PK OF 5		1.00	00006W5	ea
	(NONE)		Wick for Gimbaled Berth Lamp		5.00	00006W	ea

Updated DEV/var.

Exported model from DEV

Moved to X:\Reference\Dynamics AX\Customizations\R3\DAXKitting

Copied to C:\Temp

Installed on GOLD.

Restarted GOLD AOS (3)

Ran build script