

# Understanding the Sage 200 Database

## Sage 200 2015

**Last update:** November 2014

# CONTENTS

- Objectives of the Document..... 1
- General..... 2
  - Financial Periods ..... 2
- Sales Ledger ..... 3
  - Customer Accounts ..... 3
  - Pending Customer Accounts ..... 5
  - Customer Transactions..... 6
- Purchase Ledger ..... 9
  - Supplier Accounts..... 9
  - Pending Supplier Accounts..... 11
  - Supplier Transactions ..... 12
- CashBook..... 15
  - CashBook Accounts ..... 15
  - Cash Book Transactions..... 16
- Nominal Ledger ..... 17
  - Nominal Accounts..... 17
  - Pending Nominal Accounts..... 19
  - Nominal Transactions ..... 20
  - Nominal Journals ..... 21
  - Nominal Budgets and Actuals..... 22
  - Financial Statement Layouts..... 23
- Sales Order Processing..... 24
  - SOP Orders ..... 24
  - SOP Allocations..... 25
  - SOP Despatches ..... 27
  - SOP Invoices ..... 29
  - SOP Payments ..... 31
  - SOP Quotations..... 32
  - SOP Pro Formas ..... 32
  - SOP Repeat Orders..... 32
- Purchase Order Processing..... 33
  - POP Orders ..... 33
  - POP Receipts ..... 35
  - POP Invoices ..... 37
- Stock ..... 39
  - Item Details..... 39
  - Search Categories ..... 41

Stock Locations .....	42
Units of Measure.....	44
Stock levels.....	48
Traceable Items .....	51
Price Book .....	54
Prices.....	54
Discounts.....	55
Bill of Materials module .....	57
BOMs.....	57
BOM Allocations .....	58
BOM Builds.....	59
Project Accounting.....	60
Projects.....	60
Project Structure .....	61
Integration and Display Feature Settings.....	62
Additional Information .....	63
Project Transactions .....	64
Billing .....	66
Timesheets and Expenses .....	67
People and Resources .....	67
Expense Claims.....	68
Timesheet Entry.....	69
Project and Resource filtering.....	70
Look-up Tables.....	71
AuthorisationStatus.....	71
BackToBackStatus .....	71
BLAmountOrPercentage.....	71
BLBillHeaderStatusType.....	72
BLBillLineType.....	72
BLDetailOrSummaryType .....	72
BLDiscountOrSurcharge .....	72
BLEventStatusType .....	73
BomBuildPackageType .....	73
BomBuildSessionStatus.....	73
BomBuildSessionType.....	73
BomComponentLineType .....	74
BomCostType .....	74
BomOperationLineType.....	74
BomVersionStatus .....	75

CBAccountLocationType .....	75
CBAccountType .....	75
CBDirectDebitDocumentType .....	76
CBDirectDebitTranType .....	76
CBPaymentFrequency .....	77
CBTranType .....	77
ConfirmationIntentType .....	77
CostingMethod .....	78
CustomerType .....	78
DefaultPricingSourceType .....	78
DiscountType .....	78
DocumentPrintStatus .....	79
DocumentStatus .....	79
EntrySource .....	79
EventLogType .....	80
IntrastatDateType .....	80
IntrastatDestinationType .....	80
IntrastatEntryHeaderStatus .....	80
IntrastatStatus .....	81
INVCustomerUnitPricePref .....	81
INVInvoiceCreditType .....	81
INVInvoiceStatus .....	81
LandedCostsType .....	82
ModeOfTransport .....	82
MovementBalanceType .....	82
NatureOfTransactionCode .....	83
NLAccountReportType .....	84
NLAccountType .....	84
NLAnnualBudgetApportProfile .....	84
NLAnnualBudgetTitle .....	84
NLAnnualBudgetType .....	85
NLFinancialReportLineType .....	85
NLJournalTemplateType .....	85
NLJournalTranTaxType .....	86
NLJournalType .....	86
NLLayoutDebitOrCreditType .....	86
NL LayoutPositionType .....	86
NLNominalTranType .....	87
NLPendNominalAccountType .....	87

NominalUsage .....	87
OrderReturnLineType .....	88
PaymentType.....	88
PCAnalysisFieldType.....	88
PCBillStatus.....	89
PCBudgetLevel.....	89
PCConfigurableTerm .....	89
PCCreateProjectPreference .....	90
PCDisplayFeatureGroup.....	90
PCEntryPostingStatus .....	90
PCEntryType .....	90
PCFinancialClassification .....	91
PCOverheadUpliftModifier .....	91
PCPriceModifierType.....	91
PCPricingRuleMethodType.....	91
PCProjectArchiveStatus.....	92
PCProjectEntryDescriptor .....	92
PCProjectItemStatus .....	94
PCProjectItemType.....	94
PCProjectLevelBillingMethod.....	94
PLAllocationType .....	95
PLPaymentDocumentTypeId .....	95
POPAuthAuditEventType.....	96
POPAuthPrincipleContentType.....	96
POPAuthRoleType .....	96
POPInvoiceCreditType .....	97
POPOrderGenerationType .....	97
POPOrderReturnLineAddrType .....	97
POPOrderReturnType .....	97
POPrcptRtnLineSource .....	98
POPReceiptReturnType .....	98
POPReorderLineStatus .....	98
POPToReorderItemStatus.....	98
PSCondOperatorType .....	99
PSLogicalOperatorType.....	99
ShowOnPickingListType .....	99
SLAllocationType.....	100
SLDateToUseForAgeing.....	100
SLDocumentMessageUsedFor.....	100

SLOfficeType .....	101
SLReportMessageType .....	101
SLTrantoShowOnStatements .....	101
SOPDespatchReceiptType .....	101
SOPInvoiceCreditType .....	102
SOPOrderEntryType .....	102
SOPOrderFulfilmentMethod .....	102
SOPOrderReturnType .....	103
SOPPartialAllocationType .....	103
SOPRepeatOrderType .....	103
SourceAreaType .....	104
STKAutoGenerationOptionType .....	104
STKAutoGenerateSeparator .....	104
STKFulfilmentSequenceType .....	105
STKLabelPrintingOptionType .....	105
STKShelfLifeType .....	105
StockItemStatus .....	105
StockItemType .....	106
StockTake Selector .....	106
StockTake Status .....	106
StockTakeType .....	106
SYSAccountMemoType .....	107
SYSAccountType .....	107
SYSAddressContactUse .....	107
SYSAuditTrailEntryType .....	108
SYSCompanyLocationType .....	109
SYSCondOperatorType .....	109
SYSContactType .....	109
SYSDocTransmissionMethod .....	110
SYSDocumentType .....	110
SYSExchangeRateAction .....	111
SYSExchangeRateAmendType .....	111
SYSExchangeRateType .....	112
SYSImportType .....	112
SYSLogicalOperatorType .....	112
SYSModule .....	112
SYSModulePostingStatus .....	113
SYSNotificationPriorityType .....	113
SYSNotificationStatusType .....	113

SYSPaymentTermsBasis.....	114
SysPeriodAction .....	114
SYSPeriodActionType .....	114
SYSTaxECTerm .....	115
SYSTaxPeriodStatusType .....	115
SYSTaxTranType .....	115
SYSTraderGenerateReasonType.....	116
SYSTraderLocationType.....	116
SYSTraderRevalAllocType .....	116
SYSTraderTranType.....	116
SYSTraderWriteOffEntryType.....	117
SYSTranDateValidationClass .....	117
SYSTranDateValidationType .....	117
TermsOfDelivery .....	118
TimeUnit .....	118
TraceableItemStatus.....	119
TraceableTransType.....	119
TraceableType.....	120
TransactionGroup .....	120
TransactionType .....	120
TSActivitySearchKey .....	121
TSCategoryCompStatusType .....	121
TSCategoryCompType .....	122
TSClaimRecordStatusType .....	122
TSConfigurableTerm .....	122
TSDayOfWeek.....	123
TSExpenseClaimStatusType .....	123
TSExpensesPostingStatusType .....	123
TSExpensesPostingStatusTypeID.....	123
TSPaymentFrequencyType .....	124
TSPaymentFrequencyTypeID.....	124
TSPaymentMethod .....	124
TSPaymentMethodID .....	124
TSPostingMethodType .....	124
TSPostingMethodTypeID.....	124
TSReceiptAttachedType .....	124
TSReceiptAttachedTypeID .....	124
TSResourceHierarchyItemType.....	125
TSResourceHierarchyTypeID .....	125

TSResourceStatusType.....	125
TSResourceStatusTypeID .....	125
TSResourceType.....	125
TSResourceTypeID .....	125
TSSubmissionFrequencyType.....	125
TSSubmissionFrequencyTypeID .....	125
TSTimesheetPostStatusType .....	125
TSTimesheetPostStatusTypeID.....	125
TSTimesheetRecordStatusType.....	126
TSTimesheetRecordStatusTypeID .....	126
TSTimeUnitType.....	126
TSTimeUnitTypeID .....	126
TSUserSearchKey .....	126
TSUserSearchKeyID .....	126
TSUserStatus .....	126
TSUserStatusID.....	126
TSWorkPeriod .....	127
TSWorkPeriodID.....	127
UnitOfMeasureType.....	127
WarehouseType .....	127
WopComponentLineType .....	127
WopOrderType .....	128



## Objectives of the Document

This document aims to provide information to assist Developers and Technical Support staff working with Sage 200.

It describes the database tables relating to the main entities of the system; particularly where there are complex table relationships.

**Note:** This document is not designed to cover the entire Sage 200 database.

This document includes details about Sage 200 Accounts, Project Accounting and Time and Expenses.

Throughout this guide, there are references to fields, database tables and the interface. They are colour-coded to make it easier to understand. These are as follows:

Primary and Foreign key fields

Relationships between tables

Important non-key fields in tables

References to the application interface

[Hyperlinks to other areas in the document](#)

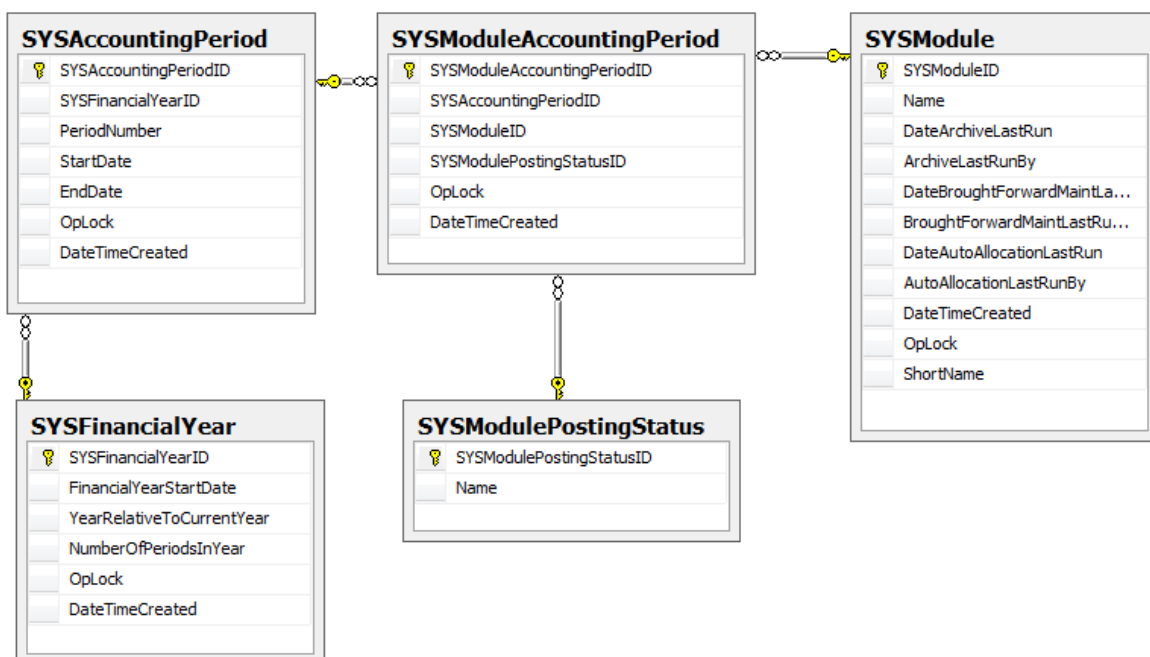
# General

## Financial Periods

The Financial accounting periods set up through [Accounting System Manager > Accounting Periods](#) relate to the modules Cashbook, Sales, Purchase, Stock and Nominal.

The period details are held in SYSAccountingPeriod records that will each link to **one** SYSFinancialYear record on [SYSFinancialYearId](#).

For each of these 3 financial modules there will be several SYSModuleAccountingPeriod records linking to **one** SYSModule with SYSModuleIds: 1 - Sales Ledger, 2 – Purchase Ledger, 3 – Nominal Ledger, 6 – Stock Control and 10 – Cash Book. There will be one SYSModuleAccountingPeriod for every SYSAccountingPeriod set up. Each SYSModuleAccountingPeriod will point to **one** [SYSModulePostingStatus](#) on [SYSModulePostingStatusId](#) that indicates the posting status for that period for that module.



# Sales Ledger

## Customer Accounts

In the application this data can be seen through [Sales Ledger > Sales Enquiries > Accounts](#).

The SLCustomerAccount record is the main customer entity record.

This will link to [one SLOfficeType](#) on [SLOfficeTypeId](#) (links to SLCustomerAccount.SLAssociatedOfficeTypeId) which controls how statements are produced for the account. If the office type is 1 – Branch, then the SLCustomerAccount.AssociatedHeadOfficeAccountId will link to the SLCustomerAccount.SLCustomerAccountId for the sales account that is the head office, as visible on the [Settings and Defaults](#) tab.

The SLCustomerAccount may also link to [one SLFinanceCharge](#) on [SLFinanceChargeId](#) which gives details of any finance charges to be levied on overdue debts. However not all accounts will have this link. This is shown on the [Credit](#) tab.

Each account may have [one or more](#) SLAccountMemo and SLAccountAttachment records linked on [SLCustomerAccountId](#), but there may be none of these linked to an account. These are visible on the [Attachments](#) and [Memo](#) tabs.

**Note:** The Analysis codes, associated with a customer, are fields held in the SLCustomerAccount table and visible on the [Analysis Codes](#) tab.

The account will link to [one or more](#) SLCustomerPeriodValue records on [SLCustomerAccountID](#) which will each link to [one](#) SLAnalysisPeriod. These trading analysis periods are set up for the Sales ledger as a whole. The SLCustomerPeriodValue records hold various monetary values relating to the customer for a particular trading period. The customer's yearly figures are held in the SLCustomerYearValue table linked to the SLCustomerAccount and to one SYSFinancialYear. These figures can be seen on the [Turnover by Period](#) and [Turnover by Year](#) tabs.

**Note** Trading analysis periods and financial periods are separate entities. These may be set up for the same or different time periods.

The SLCustomerDocument records relate to the different [document types](#) produced by the system and link with either [one](#) SLCustomerContact on [SLCustomerContactId](#) or one SLCustomerLocation on [SLCustomerLocationId](#), depending on the SLCustomerDocument.[SYSDocTransmissionMethodId](#). If this field is 1 (Email) then the record will link to [one](#) SLCustomerContact record containing an email address, if it is 0 then this record will link to [one](#) SLCustomerLocation record containing a full postal address. The link via SLCustomerContactRole to SYSTraderContactRole is a means to specify which contacts should receive documents and how they are to be delivered i.e. e-mail, fax or post. All this is visible on the [Contacts, General Information](#) and [Stationery](#) tabs.

**Note:** There may be SLCustomerContact and SLCustomerLocation records that link to [one](#) SLCustomerAccount record on [SLCustomerAccountId](#) that do not have links with SLCustomerDocument records.

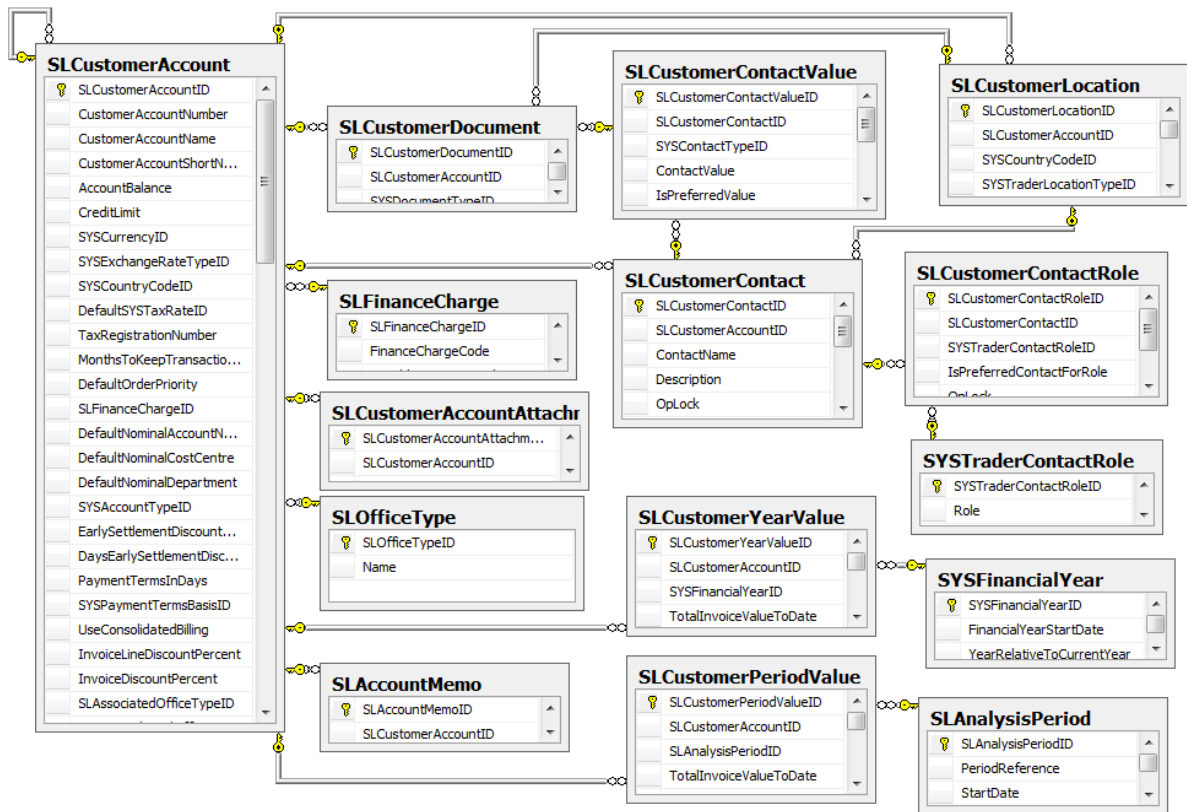


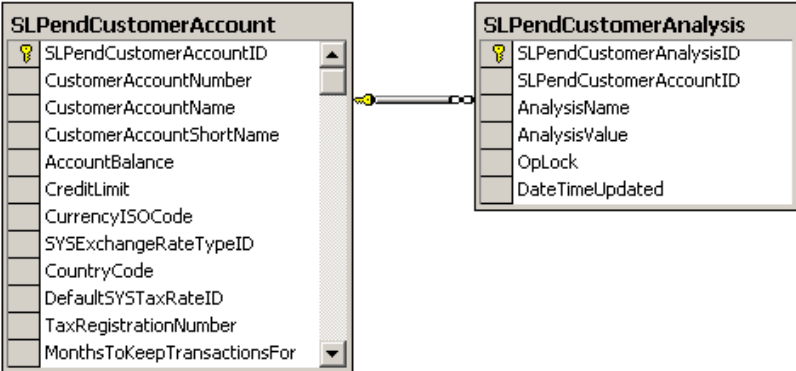
Diagram: Customer Details

## Pending Customer Accounts

Facilities to add new customer account records, without using the normal manual input routines, exist within the Sage 200 system. One method is to write records directly into special tables that can then be validated by the system and used to update the ledger if valid.

The data is held in SLPendCustomerAccount records each of which will link to **one or many** SLPendCustomerAnalysis records on **SLPendCustomerAccountID**. These records do not have any direct link to existing customer accounts as they are for new customers only – i.e. they cannot be used to update existing customer records.

**Note:** These tables should only be used by Business Partners who are accredited Sage 200 developers.



## Customer Transactions

### Posted Transactions

These are transactions that have been updated to the Sales Ledger and can be viewed using [Sales Ledger > Sales Enquiries > Current Transactions/Archived Transactions](#). There are two types of posted transaction, Current and Archived.

- Archived transactions are in the SLHistoricalCustomerTran table. Each SLCustomerAccount links to **many** SLHistoricalCustomerTran records on [SLCustomerAccountId](#).

**Note** There may be no linking archived transactions.

- For current transactions the SLCustomerAccount record will link to **many** SLPostedCustomerTran records on [SLCustomerAccountId](#).

A SLPostedCustomerTran may link to **one or many** SLTranMemo records on [SLPostedCustomerTranId](#) containing any memos created for the transaction as can be seen in the [Transactions Memo](#) tab. However there may be cases where there may be none of these records.

Each current transaction may link to **one or many** SLRevalAllocationTran records on [SLPostedCustomerTranId](#). These records contain a history of allocations and revaluations that have been applied to the posted transaction and are used for retrospective aged debtors reports among other things.

**Note:** There may be none of these transactions linking to the posted transaction, if it has not yet been allocated or revalued.

If the posted transaction has been fully or part allocated then the SLPostedCustomerTran record will link to **one or more** SLAllocationTran records on [SLPostedCustomerTranId](#). Each of these in turn link to **one** SLAllocationHeader record on [SLAllocationHeaderId](#). There will be more than one SLAllocationTran record if the posted transaction has been part allocated in more than one allocation session:

For example: An invoice dated 1/1/2005 has been part paid by a receipt dated 20/1/2005 this allocation was done on 22/1/2005. The rest of the invoice was paid by a receipt dated the 30/1/2005 this was allocated on the system on 1/2/2005. There would be two SLAllocationTran records linked to the SLPostedCustomerTran record for the invoice, but only one linking to the SLPostedCustomerTran record for each receipt.

Another way to look at this is: the SLAllocationHeader will link to **many** different SLAllocationTran records on [SLAllocationHeaderId](#) that each link to **one** SLPostedCustomerTran record, indicating that all these posted transactions are allocated together. The allocation header record is the 'group' record for the allocation session in which 2 or more posted transactions were allocated together.

This provides the allocation drill down information viewable through the [Allocation History](#) and [Allocation Details](#) tabs.

**Note:** The [Nominal](#), [Tax](#) and [Bank](#) analysis tabs use the URN from the SLPostedCustomerTran. However, this is not a physical link within the database.

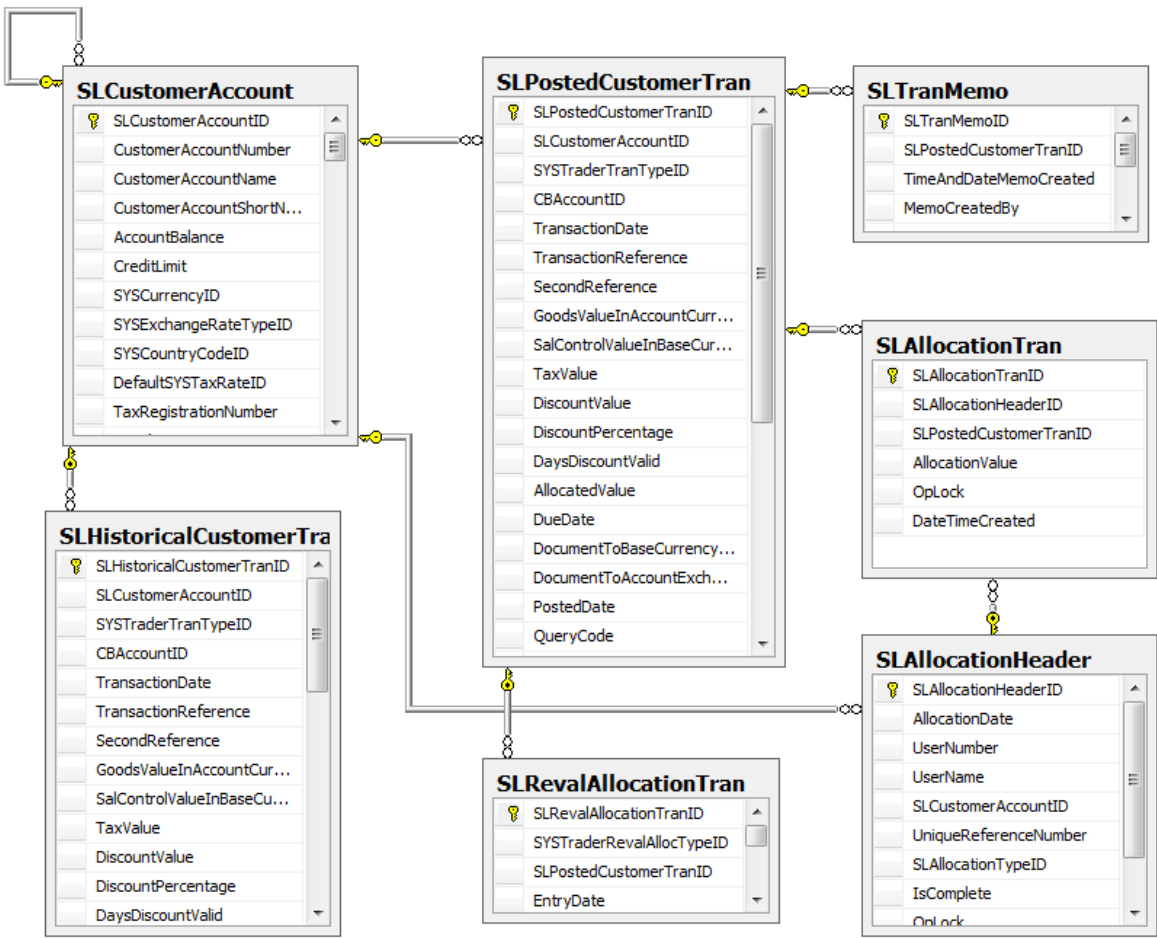


Diagram: Customer Posted Transactions

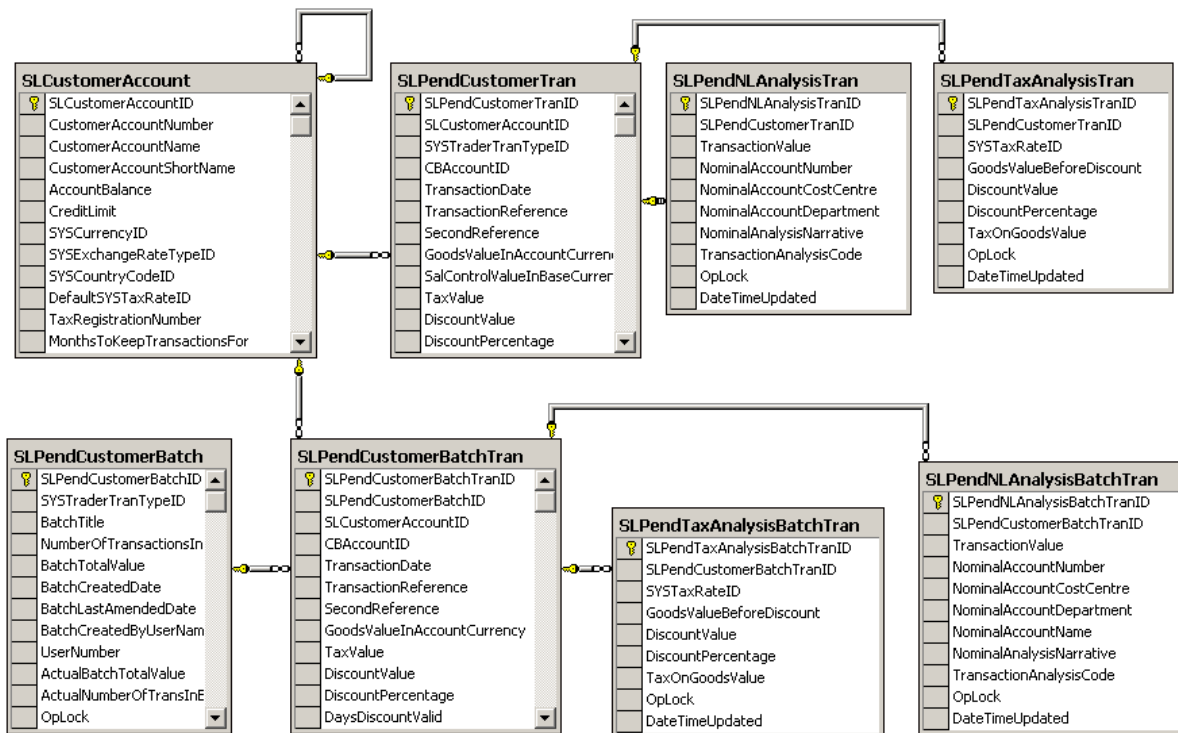
## Pending Transactions

There are two types of pending customer transactions: those entered in batches that have not yet been committed to the ledger, viewable through [Sales Ledger > Enter Transactions > Batched Transactions](#); and those posted to the Sales Ledger from other modules that are still in the pending file, viewable through [Sales Ledger > Period End Routines > Report Pending Transactions](#).

- Those in held batches are in SLPendCustomerBatchTran records linked to **one** SLCustomerAccount on **SLCustomerAccountID**. Each of these will also link to **one** SLPendCustomerBatch record on **SLPendCustomerBatchID**, this record contains details relating to the whole batch of transactions and will therefore link to **many** SLPendCustomerBatchTran records linking to different customers.

A SLPendCustomerBatchTran will have **one or many** SLPendNLAnalysisBatchTran records linked on **SLPendCustomerBatchTranID** containing the details of the nominal breakdown for the batch transaction, and to **one or many** SLPendTaxAnalysisBatchTran linked on **SLPendCustomerBatchTranID**, containing the tax analysis for this batch transaction.

- Those in the pending file are held in SLPendCustomerTran records linked to one SLCustomerAccount on **SLCustomerAccountID**. Each of these will link to **one or many** SLPendNLAnalysisTran and to **one or many** SLPendTaxAnalysisTran on **SLPendCustomerTranID**, for details of the nominal and tax analysis for this pending transaction.





# Purchase Ledger

## Supplier Accounts

In the application this data can be seen through [Purchase Ledger > Purchase Enquiries > Accounts](#).

The PLSupplierAccount record is the main supplier entity. This may link to **one or more** PLSupplierDocument records that relate to the different [document types](#) produced by the system and link with either **one** PLSupplierContact on [PLSupplierContactId](#) or **one** PLSupplierLocation on [PLSupplierLocationId](#), depending on the PLSupplierDocument. [SYSDocTransmissionMethodId](#). If this field is 1 (Email) then the record will link to **one** PLSupplierContact record containing an email address, if it is 0 then this record will link to **one** PLSupplierLocation record containing a full postal address. These details are viewed on the [Contacts](#) and [General Information](#) tabs.

**Note:** There may be PLSupplierContact and PLSupplierLocation records linking to **one** PLSupplierAccount record on [PLSupplierAccountId](#) that do not have links with PLSupplierDocument records.

Each supplier account will link to **one** PLPaymentGroup on [PLPaymentGroupId](#) which controls how payments are made to this supplier. It may also link to **one** PLFactorHouse on [PLFactorHouseId](#). These can be seen on the [Settings and Defaults](#) and [Factor Houses](#) tabs.

**Note:** There may be no factor house linked to accounts. Analysis codes are held on the PLSupplierAccount record itself.

Each account may have **one or more** PLAccountMemo and PLAccountAttachment records linked on [PLSupplierAccountId](#). These are shown on the [Attachments](#) and [Memo](#) tabs. However there may be no records linked to an account.

The account will link to **one or more** PLSupplierPeriodValue records on [PLSupplierAccountID](#) which will each link to **one** PLAnalysisPeriod. The analysis periods are set up for the Purchase ledger as a whole. The PLSupplierPeriodValue records hold various monetary values relating to the supplier for a particular analysis period. The supplier's yearly figures are held in the PLSupplierYearValue table linked to the PLSupplierAccount and to one SYSFinancialYear. These figures can be seen on the [Turnover](#) and [Turnover by Year](#) tabs.

**Note:** Analysis periods (trading periods), and financial periods are separate entities. They may be set up for the same or different time periods.

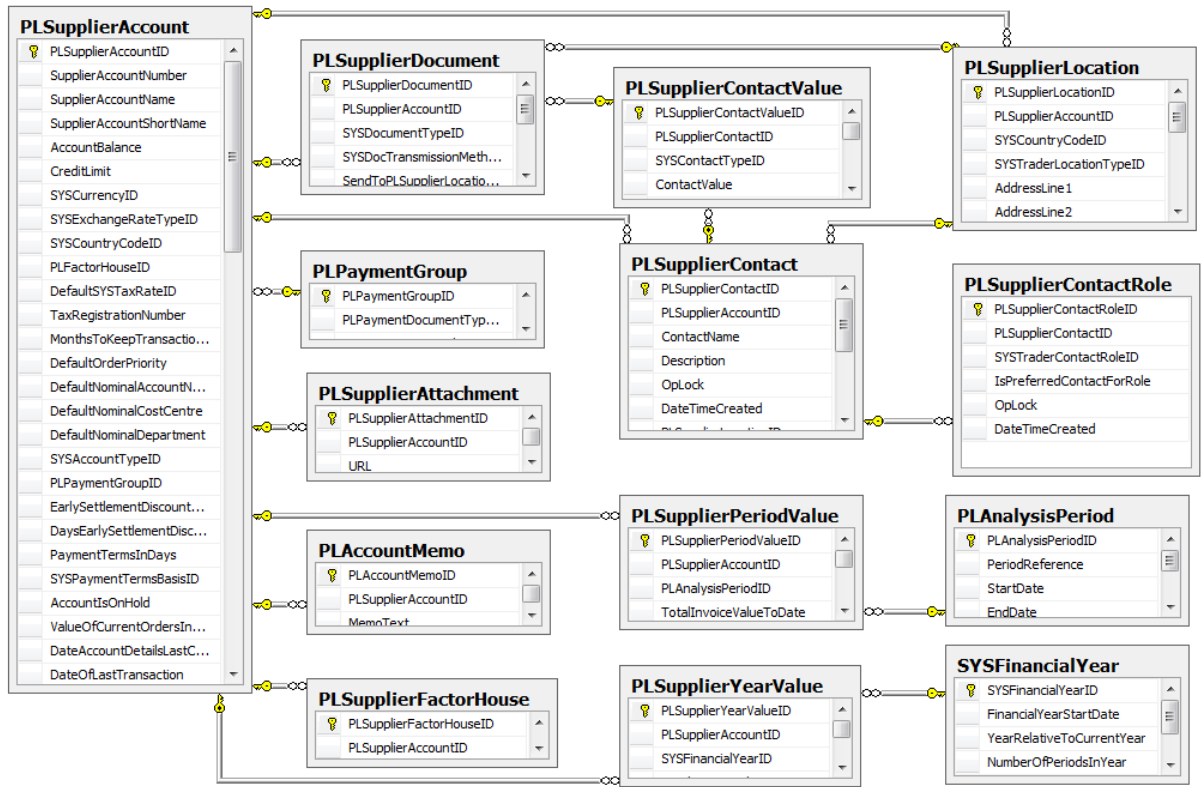


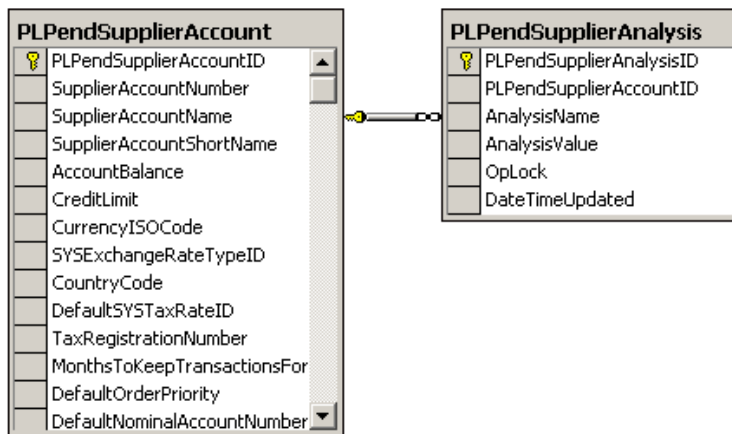
Diagram: Supplier Details

## Pending Supplier Accounts

Facilities to add new supplier account records, without using the normal manual input routines, exist within the Sage 200 system. One method is to write records directly into special tables that can then be validated by the system and used to update the ledger if valid.

The data is held in PLPendSupplierAccount records each of which will link to **one or many** PLPendSupplierAnalysis records on **PLPendSupplierAccountID**. These records do not have any direct link to existing supplier accounts as they are for new suppliers only – i.e. they cannot be used to update existing supplier records.

**Note:** These tables should only ever be used by Business Partners who are accredited Sage 200 Developers



## Supplier Transactions

### Posted Transactions

These are transactions that have been updated to the Purchase Ledger and can be viewed using [Purchase Ledger > Purchase Enquiries > Current Transactions/Archived Transactions](#). There are two types of posted transaction, Current and Archived.

- Archived transactions are in the PLHistoricalSupplierTran records. Each PLSupplierAccount links to **many** PLHistoricalSupplierTran records on [PLSupplierAccountId](#). However, there may be no linking archived transactions.
- For current transactions the PLSupplierAccount record will link to **many** PLPostedSupplierTran records on [PLSupplierAccountId](#).

A PLPostedSupplierTran may link to **one or many** PLTranMemo records on [PLPostedSupplierTranId](#).

Each current posted transaction may link to **one or many** PLRevalAllocationTran records on [PLPostedSupplierTranId](#). These records contain a history of allocations and revaluations that have been applied to the posted transaction and are used for retrospective aged creditors reports among other things.

There may be none of these linking to the posted transaction if it has not yet been allocated or revalued.

If the posted transaction has been fully or part allocated then the PLPostedSupplierTran record will link to **one or more** PLAllocationTran records on [PLPostedSupplierTranId](#). Each of these in turn link to **one** PLAllocationHeader record on [PLAllocationHeaderId](#). There will be more than one PLAllocationTran record if the posted transaction has been part allocated in more than one allocation session:

For example: An invoice dated 1/1/2005 has been part paid by a payment dated 20/1/2005. This allocation was done on 22/1/2005. The rest of the invoice was paid by a payment dated the 30/1/2005 this was allocated on the system on 1/2/2005. There would be two PLAllocationTran records linked to the PLPostedSupplierTran record for the invoice, but only one linking to the PLPostedSupplierTran record for each payment.

Another way to look at this is; the PLAllocationHeader will link to **many** different PLAllocationTran records on [PLAllocationHeaderId](#) that each link to **one** PLPostedSupplierTran record, indicating that all these posted transactions are allocated together. The allocation header record is the 'group' record for the allocation session in which 2 or more posted transactions were allocated together.

This provides the allocation drill down information viewable through the [Allocation History](#) and [Allocation Details](#) tabs.

The **Nominal**, **Tax** and **Bank** analysis tabs use the URN from the PLPostedSupplierTran but this is not a physical link within the database.

A current posted transactions may be included in the Suggested Payments list viewable through [Purchase Ledger > Period End Routines > Payment Processing > Suggested Payments](#). If so the PLPostedSupplierTran will link to **one** PLProposedPaymentPostedTran on PLSupplierPostedTranId, which will link to **one** PLProposedPayment. The PLProposedPayment can link to **one or many** SLProposedPaymentPostedTran, it contains the total of the payment that will be (or has been) made to that supplier within this suggested payments session. These records will remain linked to the posted transaction even after the payment itself has been generated, until the suggested payments list is deleted or a new one generated.

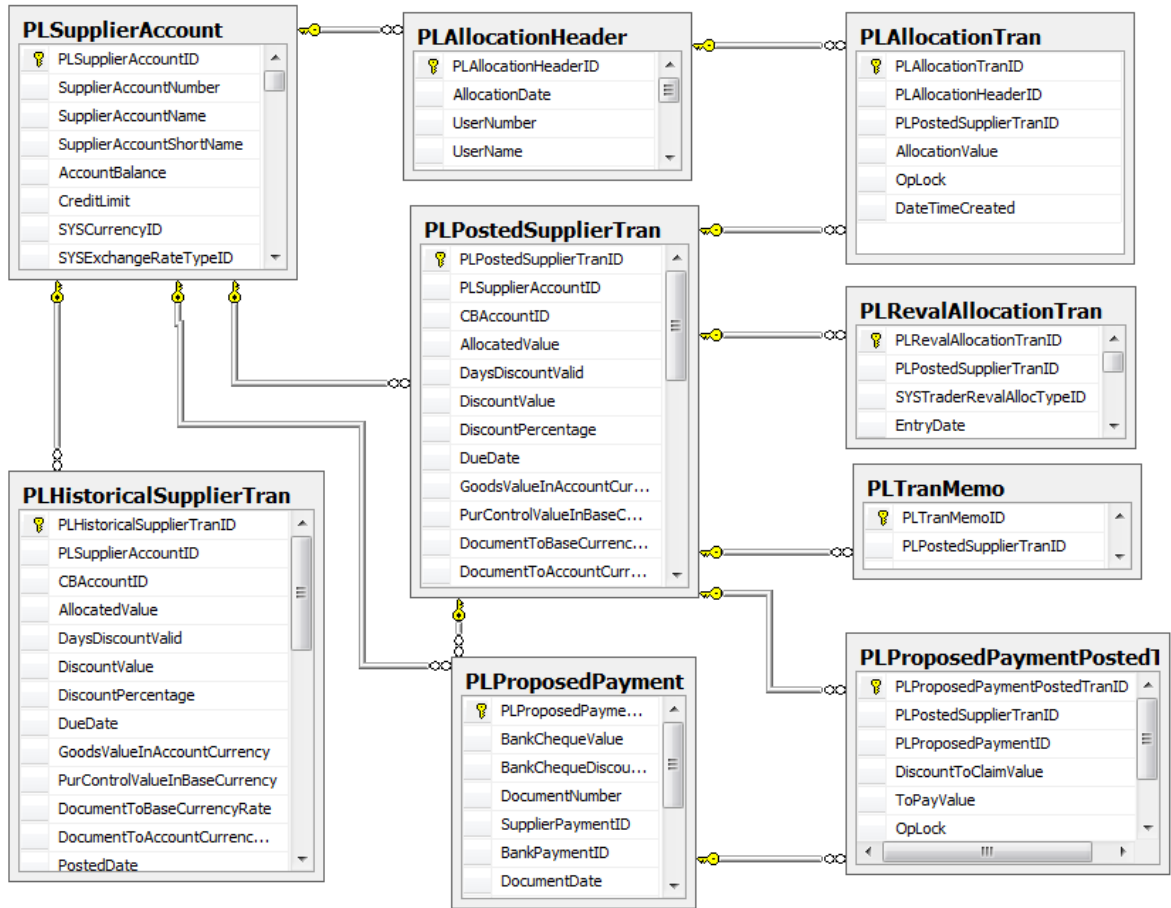


Diagram: Supplier Posted Transactions

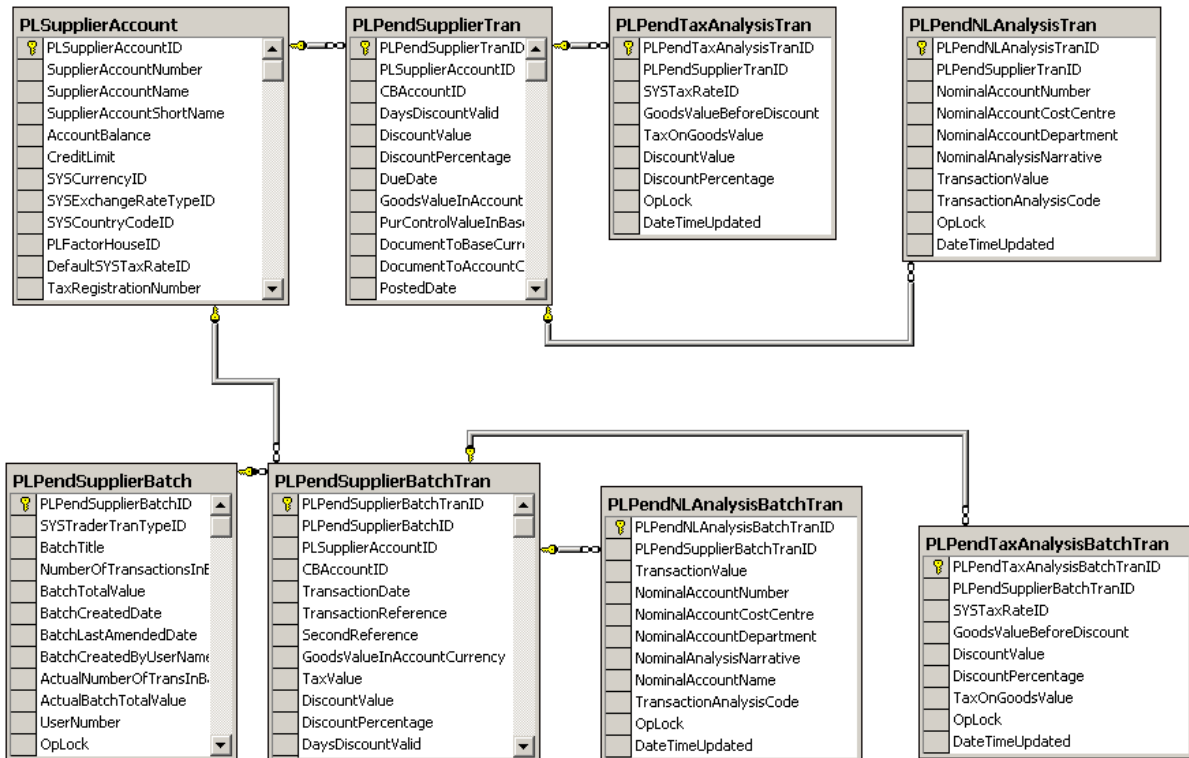
## Pending Transactions

There are two types of pending supplier transactions: those entered in batches that have not yet been committed to the ledger, viewable through [Purchase Ledger > Enter Transactions > Batched Transactions](#); and those posted to the Purchase Ledger from other modules that are still in the pending file, viewable through [Purchase Ledger > Period End Routines > Report Pending Transactions](#).

- Those in held batches are in `PLPendSupplierBatchTran` records linked to **one** `PLSupplierAccount` on `PLSupplierAccountID`. Each of these will also link to **one** `PLPendSupplierBatch` record on `PLPendSupplierBatchID`, this record contains details relating to the whole batch of transactions and will therefore link to **many** `PLPendSupplierBatchTran` records linking to different suppliers.

A `PLPendSupplierBatchTran` will have **one or many** `PLPendNAnalysisBatchTran` records linked on `PLPendSupplierBatchTranID` containing the details of the nominal breakdown for the batch transaction, and to **one or many** `PLPendTaxAnalysisBatchTran` linked on `PLPendSupplierBatchTranID`, containing the tax analysis for this batch transaction.

- Those in the pending file are held in `PLPendSupplierTran` records linked to **one** `PLSupplierAccount` on `PLSupplierAccountID`. Each of these will link to **one or many** `PLPendNAnalysisTran` and to **one or many** `PLPendTaxAnalysisTran` on `PLPendSupplierTranID`, for details of the nominal and tax analysis for this pending transaction.



# Cash Book

## Cash Book Accounts

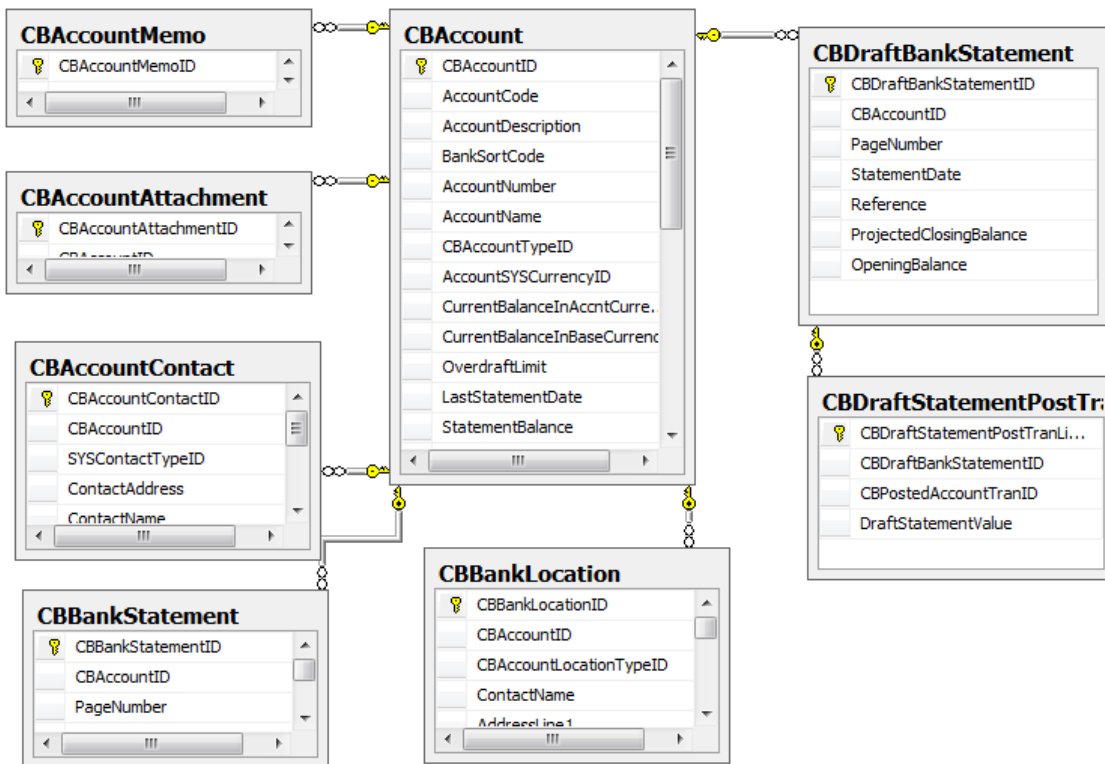
The main account data is viewable through [Cash Book > Cash Book Enquiries > Accounts](#).

Each CBAccount record may link to **one or many** CBAccountAttachment and **one or many** CBAccountMemo records on **CBAccountId**. However, there may be no linked records to an account.

The account may link to **one** CBBankLocation record on **CBAccountId**. This contains the address details entered for the account. It may also link to **one or many** CBAccountContact records on **CBAccountId** containing the email and web addresses entered.

There may be **one or many** linking CBBankStatement records linked on **CBAccountId** containing details of statements received for this account. There may also be **one or more** linking CBDraftBankStatement records for any uncompleted bank reconciliations.

**Note:** There may be none of these records linked to an account; particularly if it is not an actual bank account. For example, if the account has been set up to record your petty cash.



## Cash Book Transactions

There are 3 types of transactions that may link to an account.

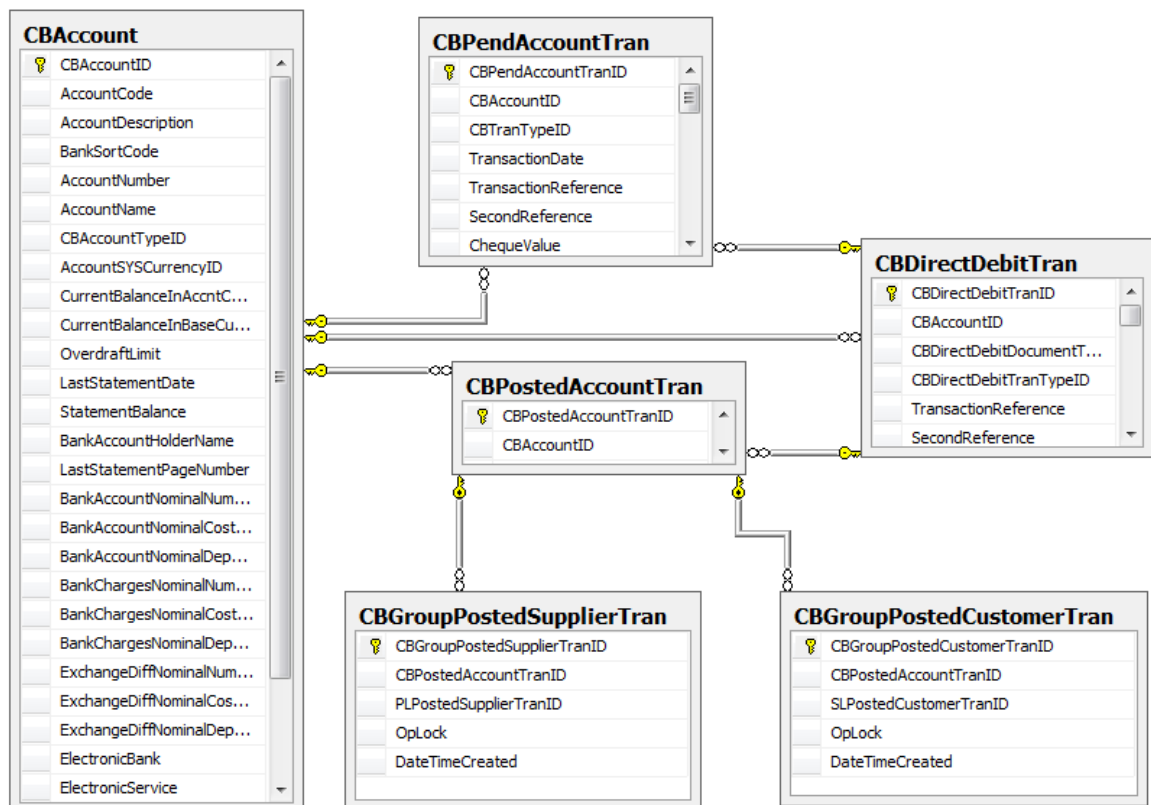
- **One or many** CBDirectDebitTran records may link on **CBAccountId** containing details of any direct debits or standing orders set up for the account. These are viewable through **Cashbook > Cashbook Enquiries > SOs and DDs**. However there may be no CBDectDebitTran records on this account.
- Transactions that have been updated to the ledger are held in CBPostedAccountTran records, the CBAccount may link to **one or many** of these on **CBAccountId**. These are viewable through **Cashbook > Cashbook Enquiries > Transactions**.

Some of these transactions may also link to **one** CBDirectDebitTran on **CBDirectDebitTranId**, if the posted transaction was generated as a result of a direct debit or standing order.

Each of these may link to a CBGroupPostedCustomerTran or a CBGroupPostedSupplierTran record on **CBPostedAccountTranId** if the transaction was entered as part of a group of cashbook transactions. These records link through to the associated Sales or Purchase transactions on **SLPostedCustomerTranId** or **PLPostedSupplierTranId**.

- Transactions that are written directly to the Sage 200 files by an external application are held in CBPendAccountTran records which link to **one** CBAccount on **CBAccountId**.

**Note:** This table should only be used by Business Partners who are accredited Sage 200 Developers.





# Nominal Ledger

## Nominal Accounts

Each nominal account, can be one of [NLAccountType](#) posting, group or memorandum. These are viewable through [Nominal Ledger > Nominal Enquiries > Accounts](#)

Each account is a combination of an account number held in the NLAccountNumber table, a cost centre held in the NLCostCentre table and a department held in the NLDepartment table.

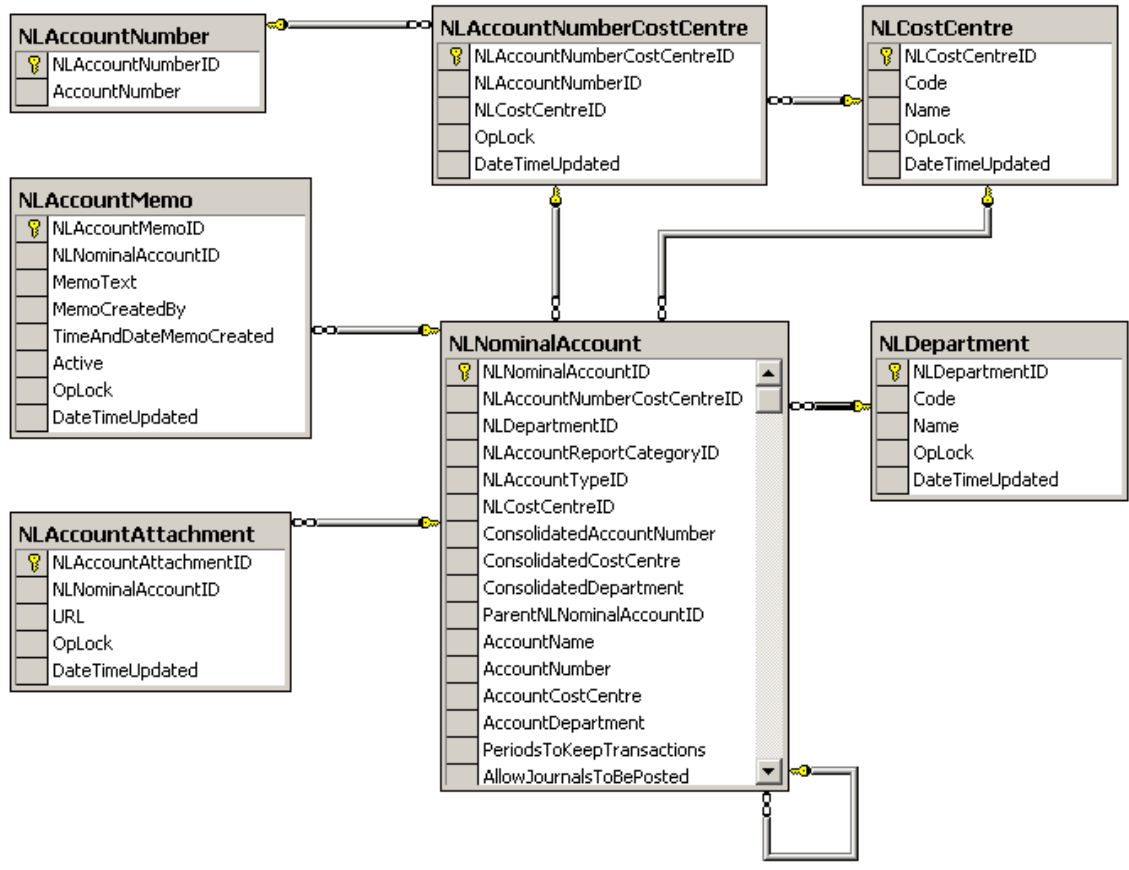
**Note:** The cost centre and department may be blank, but there will be an NLCostCentre and an NLDepartment record for a blank code.

This is a 3 way **many-to-many** relationship (an account number can link to many cost centres and departments, a cost centre can link to many account numbers and departments and departments can link to many account numbers and cost centres).

To resolve these relationships additional tables are created...

The NLAccountNumberCostCentre table resolves the **many-to-many** relationship between account numbers and cost centres (it links to **one** NLAccountNumber on [NLAccountNumberId](#) and **one** NLCostCentre on [NLCostCentreId](#)). The NLNominalAccount record itself, by linking to **one** NLAccountNumberCostCentre on [NLAccountNumberCostCentreId](#) and **one** NLDepartment on [NLDepartmentId](#) resolves that **many to many** relationship. The NLNominalAccount then is the table that holds the main details of the actual nominal accounts.

Each NLNominalAccount may have **many** NLAccountAttachment and NLAccountMemo records linked on [NLNominalAccountId](#). These are viewable using the [Attachments](#) and [Memos](#) tabs.

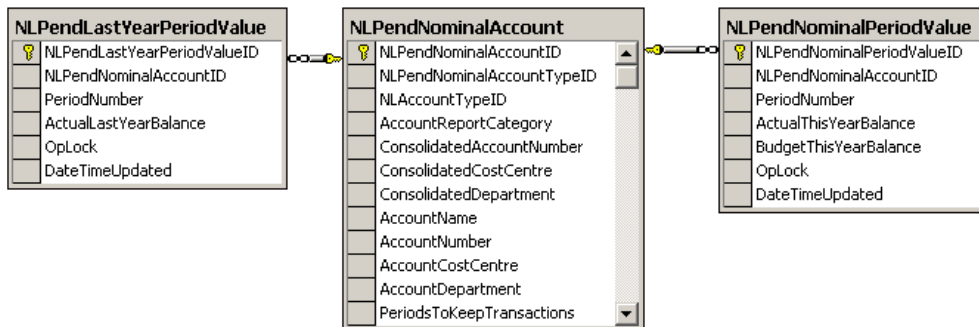


## Pending Nominal Accounts

Facilities to add new nominal account records, without using the normal manual input routines, exist within the Sage 200 system. One method is to write records directly into special tables that can then be validated by the system and used to update the ledger if valid.

The data is held in NLPendNominalAccount records each of which will link to **one or many** NLPendLastYearPeriodValue on **NLPendNominalAccountID** and to **one or many** NLPendNominalPeriodValue on **NLPendNominalAccountID**. These records do not have any direct link to existing nominal accounts as they are for new accounts only – i.e. they cannot be used to update existing nominal accounts.

**Note:** These tables should only be used by Business Partners who are accredited Sage 200 Developers



## Nominal Transactions

### Posted Transactions

These are transactions that have been updated to the Nominal Ledger and can be viewed using **Nominal Ledger > Nominal Enquiries > Current Transactions/Archived Transactions**. There are two types of posted transaction, Current and Archived.

- Archived transactions are in the NLHistoricalNominalTran records. Each NLNominalAccount links to **many** NLHistoricalNominalTran records on **NLNominalAccountID**. However there may be no linked archived transactions.
- For current transactions the NLNominalAccount record will link to **many** NLPostedNominalTran records on **NLNominalAccountID**.

The **Associated Nominal Postings** and various **Analysis** tabs use the URN from the NLPostedNominalTran but this is not a physical link in the database.

### Pending Transactions

There are two types of Nominal pending transactions (referred to as waiting postings), that can be viewed in the application through **Nominal Ledger > Period End Routines > Current Waiting Postings/Deferred Waiting Postings**.

Neither of these types has any direct link to the nominal accounts although each contains the fields AccountNumber, CostCentre and Department. This is because it is perfectly valid to post transactions for accounts that do not exist as long as the System Setting 'Update Nominal ledger immediately transactions are posted' has not been selected.

**Note:** If the setting is selected, transactions update the nominal accounts directly; i.e. they become NLPostedNominalTran records.

- Current Posting file – is made up of NLPendNominalTran records
- Deferred Posting file – is made up of NLDeferredNominalTran records.

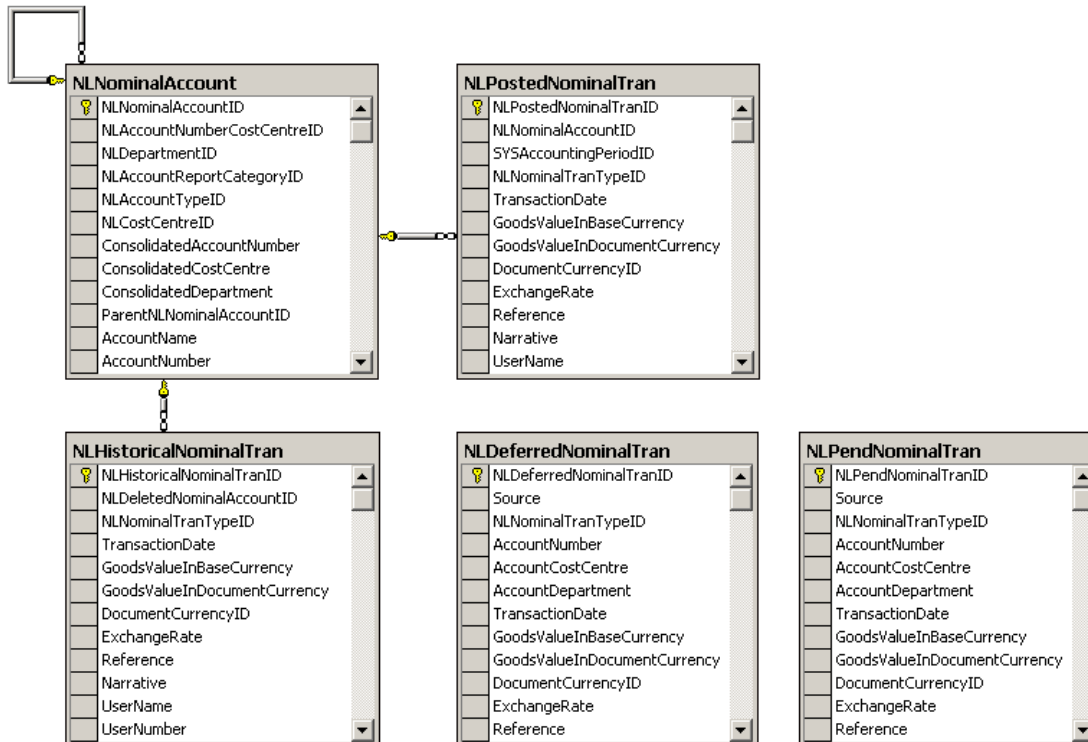


Diagram: Nominal Transactions

## Nominal Journals

When journals are entered they are usually posted straight away, and they will create different records depending on the Nominal Settings:

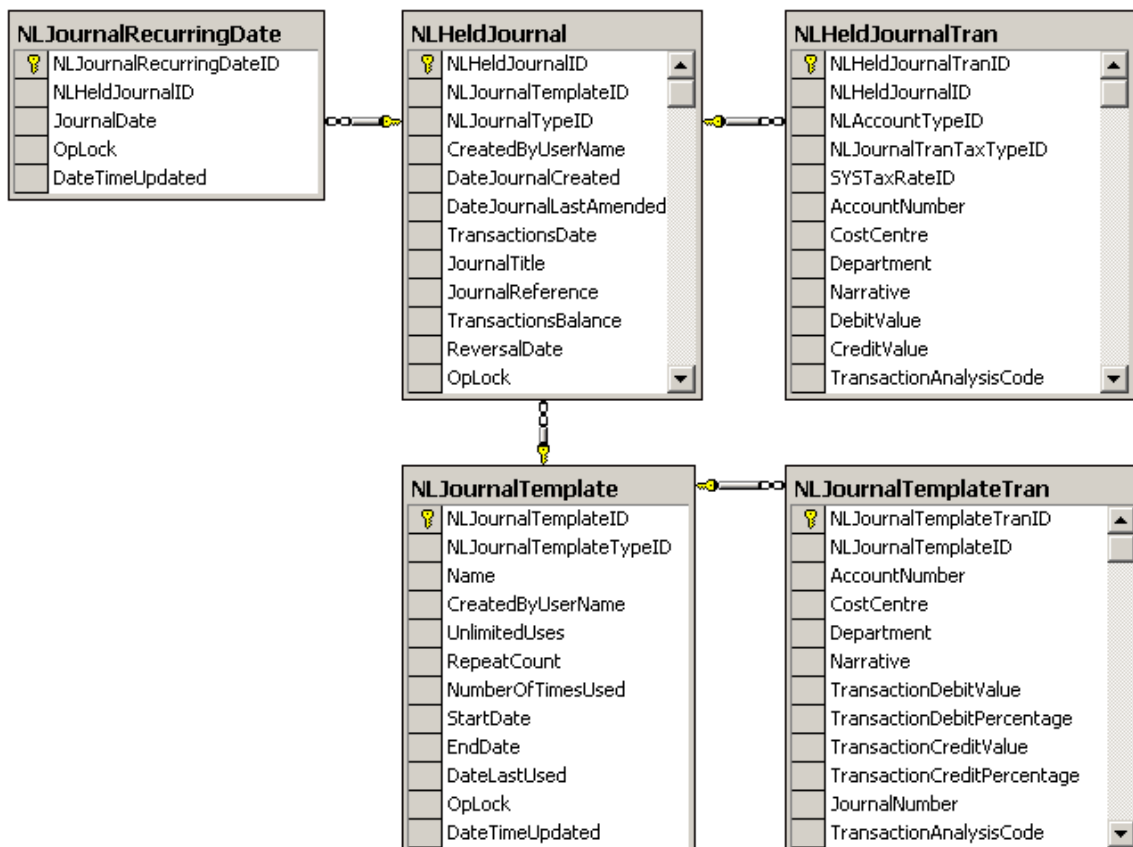
- If the setting 'Update Nominal ledger immediately journal entries are posted' is selected NLPostedNominalTran records will be created for those transactions dated earlier than (or on) the latest [Open period](#) end date.
- If the setting is not selected then transactions dated earlier than (or on) the latest Open period end date will create NLPendNominalTran records.
- Regardless of the setting transactions dated later than the latest Open period end date will create NLDeferredNominalTran records.

However journal entries can be held for posting at some time later. These can be viewed through [Nominal Ledger > Enter Transactions > Held Journals](#). In this case there will be one NLHeldJournal record that links to **one or many** NLHeldJournalTran records on [NLHeldJournalId](#).

**Note:** These are not directly linked to the NLNominalAccount records similarly to pending transactions.

The NLHeldJournal may link to **one** NLJournalTemplate record, on [NLJournalTemplateID](#), if it was created from a journal template. It may also link to **one or many** NLJournalRecurringDate on [NLHeldJournalId](#) if the journal is a recurring one.

NLJournalTemplate records viewable through [Nominal Ledger > Enter Transactions > Journal Templates](#) will link to **one or many** NLJournalTemplateTran records on [NLJournalTemplateID](#) and may link to **one or many** NLHeldJournal records.



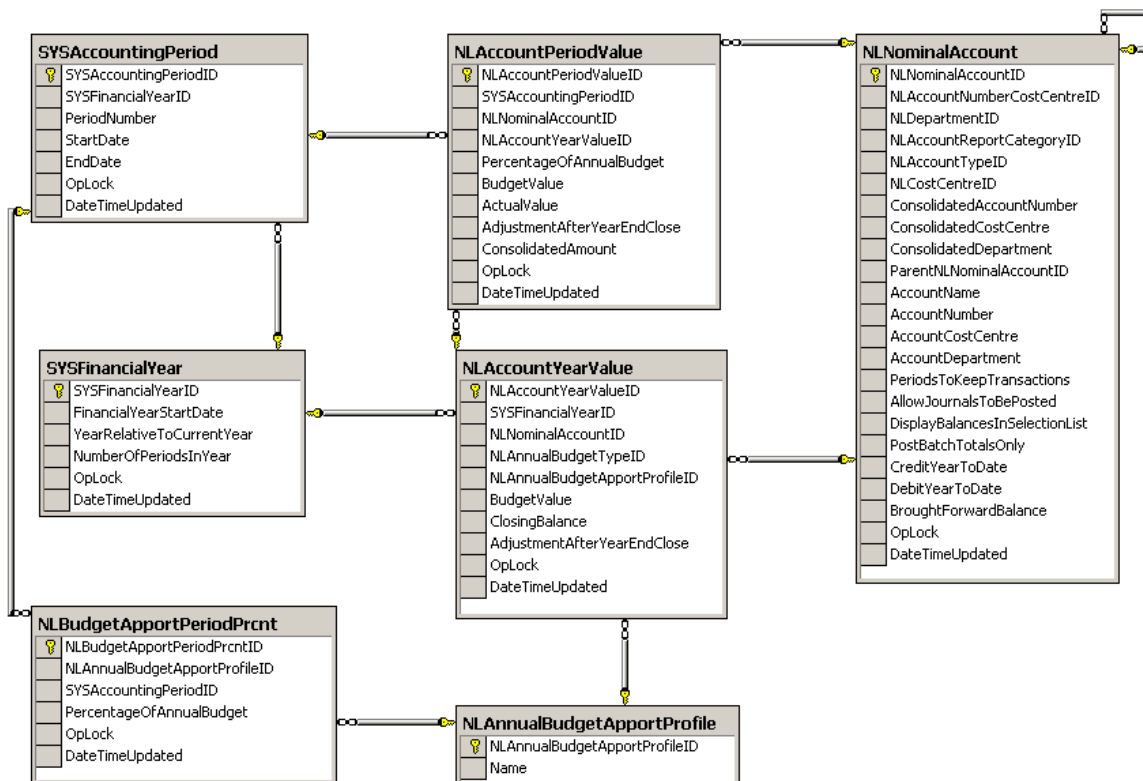
## Nominal Budgets and Actuals

Budget profiles set up through **Nominal Ledger > Utilities > Ledger Set Up > Ledger Settings** are held in **NLAnnualBudgetApportProfile** records that link to **one or many** **NLBudgetApportPeriodPrcnt** on **NLAnnualBudgetApportProfileId**. There will be one for each **SYSAccountingPeriod**, these hold the percentage of the annual budget for that period.

Each **NLNominalAccount** record will link to **many** **NLAccountYearValue** records on **NLNominalAccountID**, linking to **one** **SYSFinancialYear**. It will also link to **many** **NLAccountPeriodValue** records linking to **one** **SYSAccountingPeriod**.

If the nominal account used one of the budget profiles for its budget values then the

**NLAccountYearValue** will link to **one** **NLAnnualBudgetApportProfile** on **NLAnnualBudgetApportProfileId**.



## Financial Statement Layouts

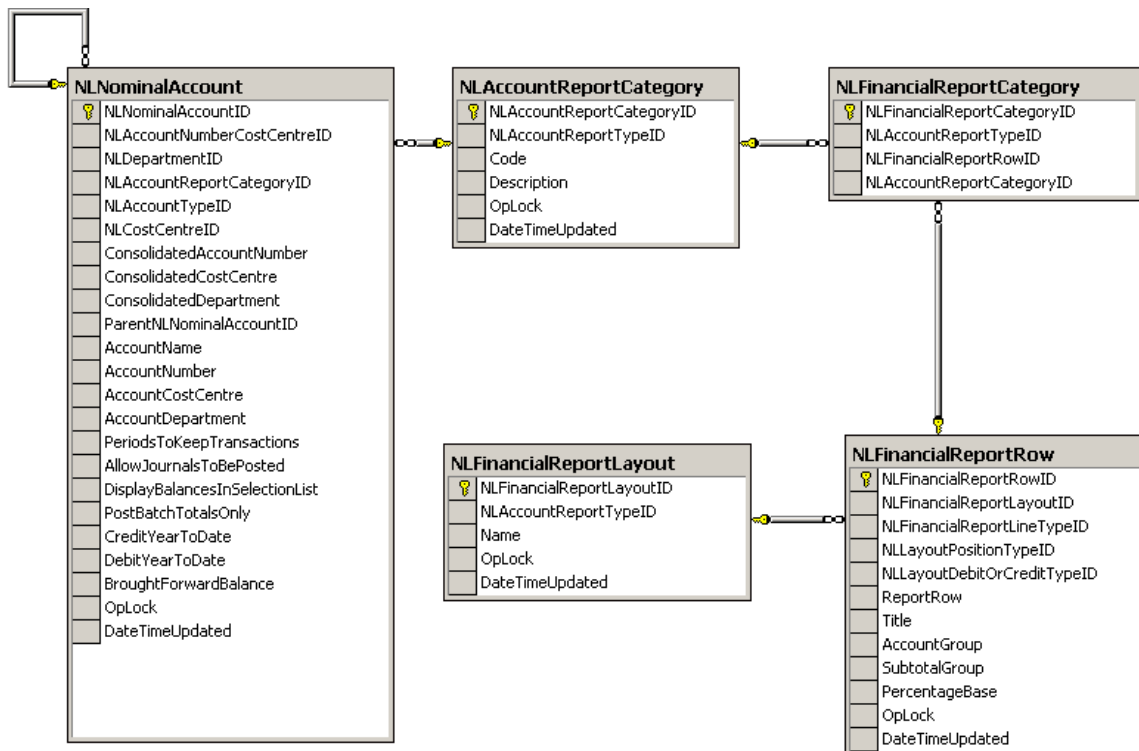
For every financial statement layout set up through **Nominal Ledger > Utilities > Ledger Set Up > Financial Statement Layouts** there will be an **NLFinancialReportLayout** record which links to **many** **NLFinancialReportRow** records on **NLFinancialReportLayoutId**.

Each **NLFinancialReportRow** links to **one or many** **NLFinancialReportCategory** records on **NLFinancialReportRowId**.

The **NLFinancialReportCategory** records set up through **Nominal Ledger > Utilities > Ledger Set Up > Report Categories** link to **one** **NLAccountReportCategory** on **NLAccountReportCategoryId** which in turn can link to **many** **NLNominalAccounts** on **NLAccountReportCategoryId**.

Each **NLNominalAccount** will link to **one** **NLAccountReportCategory** on **NLAccountReportCategoryId**.

Each report category can appear in many rows on different financial statements, and each row can contain many report categories. To split this **many to many** relationship the table **NLFinancialReportCategory** exists that links to **one** **NLAccountReportCategory**, on **NLAccountReportCategoryId** to **one** **NLFinancialReportRow** on **NLFinancialReportRowId**.



# Sales Order Processing

## SOP Orders

When an order is taken through the application the following records are set up:

- SOPOrderReturn is the order 'header' which contains information relating to the order as a whole, but not to individual order lines.
- The SOPOrderReturnLine records contain the details of each order line of the various [types](#).

The basic unallocated order will consist of **one** SOPOrderReturn, linked to **one or many** SOPOrderReturnLine records on **SOPOrderReturnID**. The order number is held in SOPOrderReturn.**DocumentNo**. The SOPOrderReturn will link to **one** SLCustomerAccount by linking SOPOrderReturn.**CustomerID** to SLCustomerAccount.**SLCustomerAccountID**

For each SOPOrderReturnLine where the **LineTypeID** is 0 (Standard) there will be **one** SOPStandardItemLink record linked on **SOPOrderReturnLineID**. This links to the StockItem on **ItemID**.

The SOPOrderReturn may link to **one** SOPPaymentMethod on **SOPPaymentMethodID**, to **one or many** SOPCancelledLine records on **SOPOrderReturnID** and to **one** SOPDocDelAddress on **SOPOrderReturnID**.

**Note:** These records do not have to exist.

The order profit figures viewable via **SOP Enquiries > Sales Documents/Sales Document Lines** and click the **Profit Analysis** button, are held the SOPOrderProfitAnalysis record linked to the SOPOrderReturn, and the SOPOrderLineProfitAnalysis record linked to each SOPOrderReturnLine.

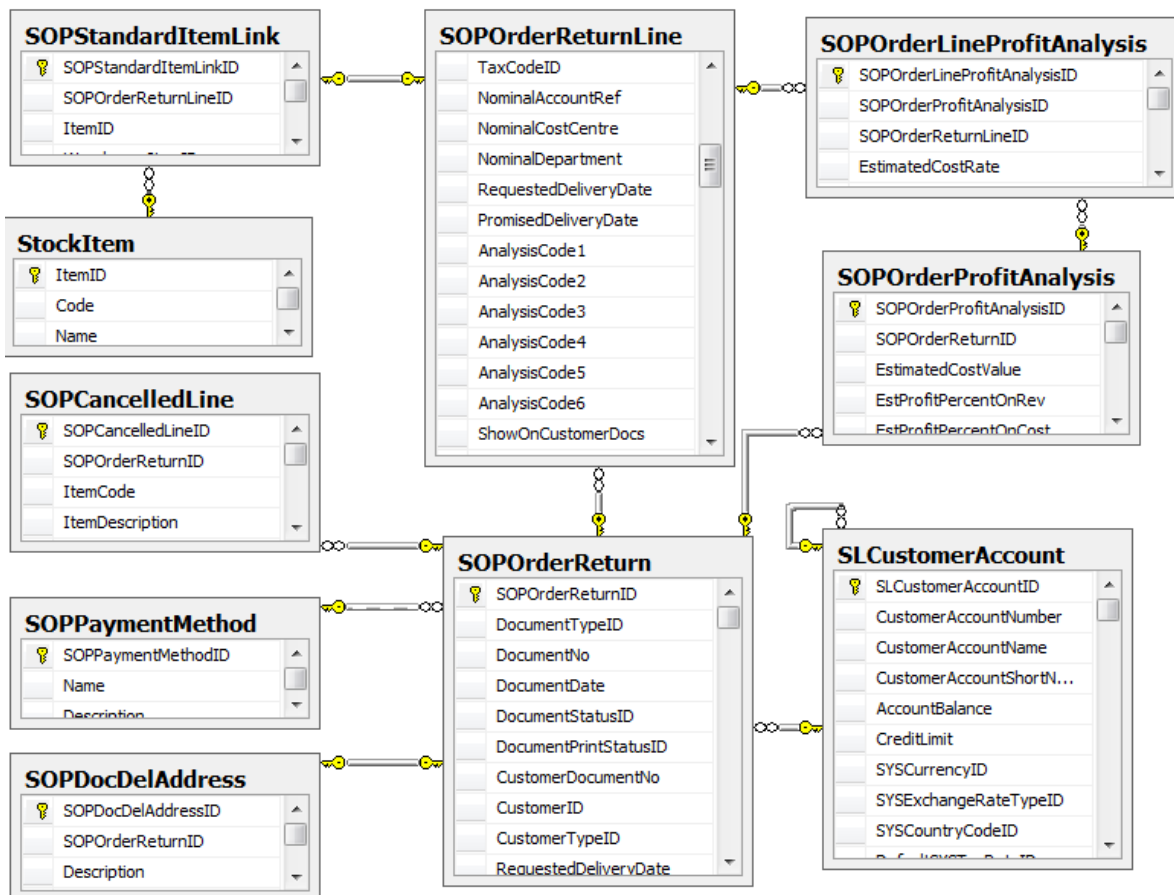


Diagram: Sales Orders



## SOP Allocations

To check the data relating to SOP allocations, viewable using the **Allocations** tab in the **Sales Document Lines** workspace:

- Find the SOPOrderReturn record using the DocumentNo
- Link the SOPOrderReturn record to its **one or more** SOPOrderReturnLine using the **SOPOrderReturnId**.

### For All Items

If the order line has a non-zero quantity in **AllocatedQuantity**, it must have additional linked records as follows:

- The SOPOrderReturnLine should link to **one or more** SOPAllocationLine records - on **SOPOrderReturnLineID** – where the total of the SOPAllocationLine.**AllocatedQuantity** fields is equal to the SOPOrderReturnLine.**AllocatedQuantity**.
- Each of the SOPAllocationLine records should have **one** matching AllocationBalance record (Stock), linked by **AllocationId** with an identical **AllocatedQuantity**.
- The BinItem relating to **one or more** AllocationBalance records, for this stock item in this bin, linked via the **BinItemId** will contain the total of the linked AllocationBalance **.AllocatedQuantity** fields in Binitem.**QuantityAllocatedSOP**.
- The WarehouseItem relating to this stock item in this warehouse, linked to **one or more** BinItem records on **WarehouseItemId**, will contain the total of the linked Binitem.**QuantityAllocatedSOP** fields in the WarehouseItem.**QuantityAllocatedSOP**.

### For Traceable Items

If the Stock setting 'Numbers are selected when allocating stock' is set then these records must also exist:

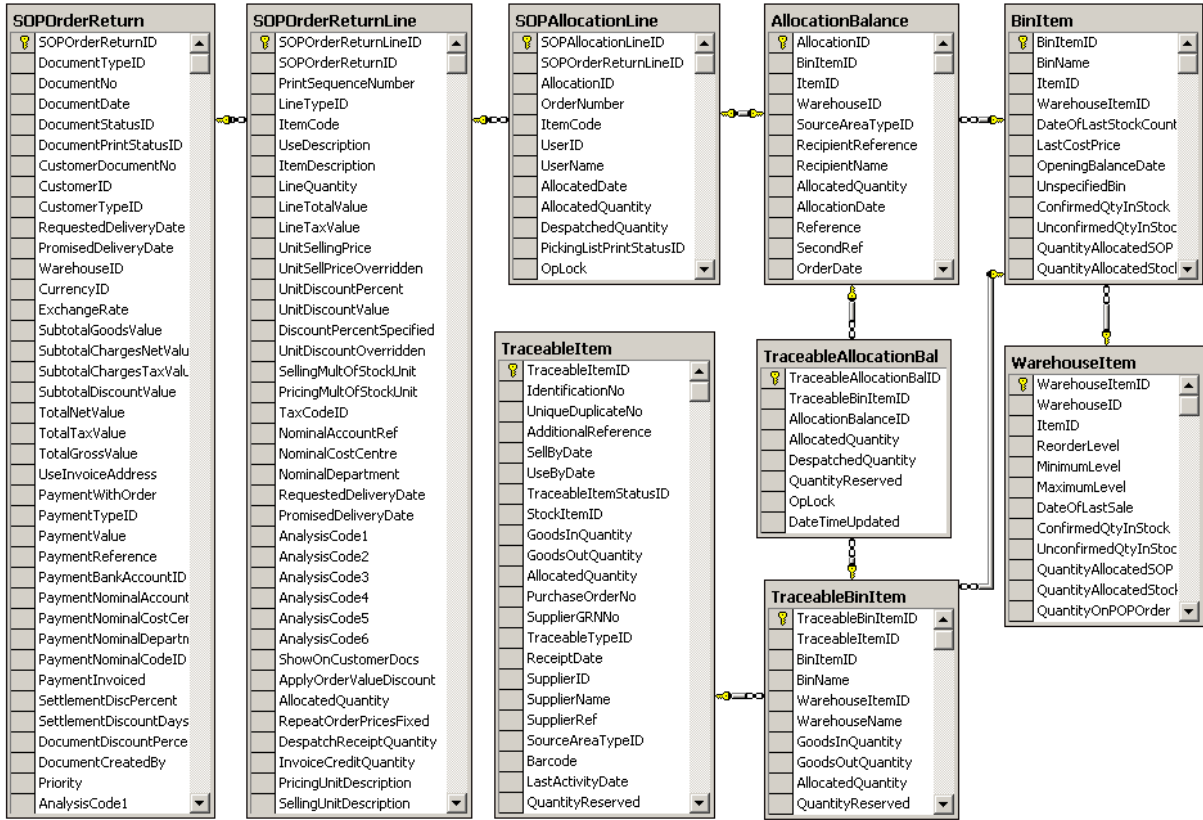
- Each AllocationBalance record must link to **one or more** TraceableAllocationBal records on **AllocationBalance Id** where the total of the TraceableAllocationBal.**AllocatedQuantity** fields will total the AllocationBalance **.AllocatedQuantity**.
- Each TraceableAllocationBal will then link to **one** TraceableBinItem via **TraceableBinItemId** where the TraceableBinitem.**AllocatedQuantity** will match the total of all TraceableAllocationBal.**AllocatedQuantity** linked to it.
- The TraceableBinItem will then link to **one** TraceableItem via **TraceableItemId**, this will give the specific batch or serial number for this SOP allocation in the TraceableItem.**IdentificationNo** field.

When viewing **Sales Document Lines** to view the allocations relating to an order line, if you click the **Traceable Allocation details** tab the information displayed is from these linked traceable records.

When using **Stock > Enquiries > Traceable Items** to view a particular batch or serial number and select the **Allocations** tab, it is these traceable records that are used to link through to the Sales Order.

**Note:** One TraceableBinitem may link to one or more TraceableAllocationBal records (if it is a batch rather than a serial item). One TraceableItem may link to one or more TraceableBinItem records (if it is a batch rather than a serial item).

The AllocationBalance.**AllocatedQuantity** will match the total of all these linked TraceableItem.**AllocatedQuantity** fields.



## SOP Despatches

To check the data relating to SOP despatches, viewable using the [Despatch Notes/Receipts](#) tab in the [Sales Document Lines](#) workspace. If any part of an order has been despatched then these records must exist.

- Find the SOPOrderReturn record using the DocumentNo.
- Then link the SOPOrderReturn record to its **one or more** SOPOrderReturnLine using the [SOPOrderReturnId](#).

### For All Items

If the order line has a non-zero quantity in [DespatchReceiptQuantity](#), it must have additional linked records as follows:

- The SOPOrderReturnLine should link to **one or more** SOPDespatchReceiptLine records - on [SOPOrderReturnLineId](#) – where the total of the SOPDespatchReceiptLine.[DespatchReceiptQuantity](#) fields is equal to the SOPOrderReturnLine.[DespatchReceiptQuantity](#).
- Each SOPDespatchReceiptLine will link to **one** SOPDespatchReceipt record linked via [SOPDespatchReceiptId](#).

**Note:** One SOPDespatchReceipt may link to **one or more** SOPDespatchReceiptLine records, not necessarily all for the same item.

- If the despatch has been invoiced the SOPDespatchReceiptLine will link to **one** SOPInvoiceCreditLine on [SOPInvoiceCreditLineId](#).

### For Traceable Items

- Each SOPDespatchReceiptLine will link to **one or more** TraceSOPDespRcptLine record on [SOPDespatchReceiptLineId](#) where the total of TraceSOPDespRcptLine.[Quantity](#) matches the SOPDespatchReceiptLine.[DespatchReceiptQuantity](#).
- Each TraceSOPDespRcptLine will link to **one** TraceableBinItem on [TraceableBinItemId](#).
- The TraceableBinItem will then link to **one** TraceableItem via [TraceableItemId](#); this will give the specific batch or serial number for this SOP despatch in the TraceableItem.[IdentificationNo](#) field.

When viewing the despatches for an order line in the application, if you select the [Traceable Despatch/Receipt Details](#) tab it is these traceable records that are used to obtain the information displayed.

When using [Stock > Traceable Items >](#) and select the [Issues](#) tab it is these traceable records that are used to link through to the Sales order.

**Note:** One TraceableBinItem may link to **one or more** TraceSOPDespRcptLine records (if it is a batch rather than a serial item). One TraceableItem may link to one or more TraceableBinItem records (if it is a batch rather than a serial item). Sales orders migrated from Line 50 or Line 100 will not have linking TraceSOPDespRcptLine records as these are impossible to create during migration.

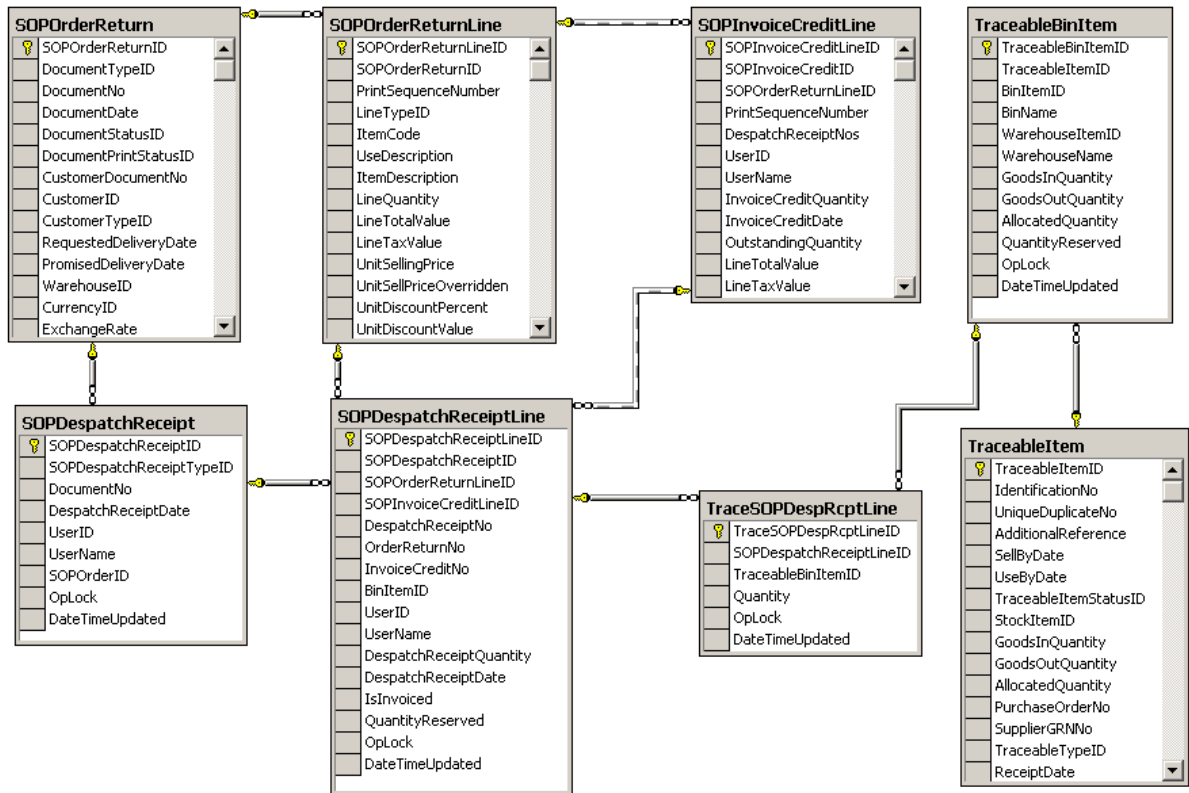


Diagram: SOP Despatches

## SOP Invoices

To check the data relating to SOP invoices, viewable using the **Invoices/Credits** tab in the **Sales Document Lines** workspace. If any part of an order has been invoiced then these records must exist.

These records can also be viewed using the **Sales Invoices and Credits** workspace which allows you to select using the invoice number.

- Find the SOPOrderReturn record using the DocumentNo (the order number displayed on screen).
- Then link the SOPOrderReturn record to its **one or more** order lines using the **SOPOrderReturnId**.

### For All Items

- The SOPOrderReturnLine should link to **one or more** SOPInvoiceCreditLine records - on **SOPOrderReturnLineId** – where the total of the SOPInvoiceCreditLine.**InvoiceCreditQuantity** fields is equal to the SOPOrderReturnLine.**InvoiceCreditQuantity**.
- Each SOPInvoiceCreditLine will link to **one** SOPInvoiceCredit record linked via **SOPInvoiceCreditId**.

If you are viewing the order line in the application and click the Invoices button it is the information from these records that is displayed.

**Note:** One SOPInvoiceCredit record may link to **one or more** SOPInvoiceCreditLine records not necessarily all for the same stock item.

- The SOPInvoiceCredit will link to **one** SOPInvCredAddress on **SOPInvoiceCreditId** containing the postal address for sending the invoice. It will also link to **one** SOPInvCredDelAddress on **SOPInvoiceCreditId**, this contains the delivery address associated with the order at the time this invoice was printed.

**Note:** An order may result in more than one invoice.

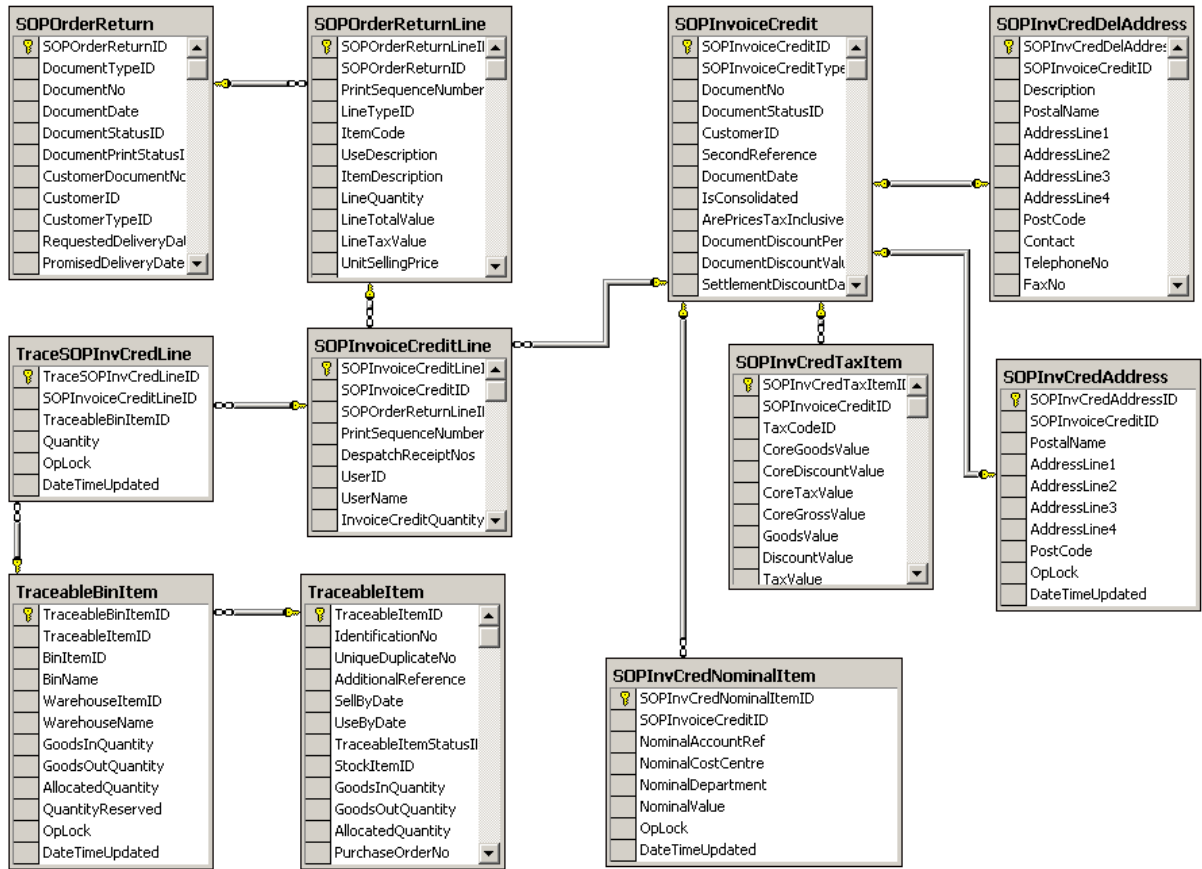
- The SOPInvoiceCredit will link to **one or many** SOPInvCredNominalItem and **one or many** SOPInvCredTaxItem, on **SOPInvoiceCreditId**, holding the nominal and tax analysis associated with the invoice.

### For Traceable Items

- Each SOPInvoiceCreditLine will link to **one or more** TraceSOPInvCredLine record on **SOPInvoiceCreditLineId** where the total of TraceSOPInvCredLine.**Quantity** matches the SOPInvoiceCreditLine.**InvoiceCreditQuantity**.
- Each TraceSOPInvCredLine will link to **one** TraceableBinItem on **TraceableBinItemId**.
- The TraceableBinItem will then link to **one** TraceableItem via **TraceableItemId**; this will give the specific batch or serial number for this SOP invoice line in the TraceableItem.**IdentificationNo** field.

When viewing the invoices for an order line in the application, if you select the **Traceable Invoice/Credit Details** tab it is these traceable records that are used to obtain the information displayed.

**Note:** One TraceableBinItem may link to one or more TraceSOPInvCredLine records (if it is a batch rather than a serial item). One TraceableItem may link to one or more TraceableBinItem records (if it is a batch rather than a serial item). Sales orders migrated from Line 50 or Line 100 will not have TraceSOPInvCredLine records as these links are impossible to create during data conversion.



## SOP Payments

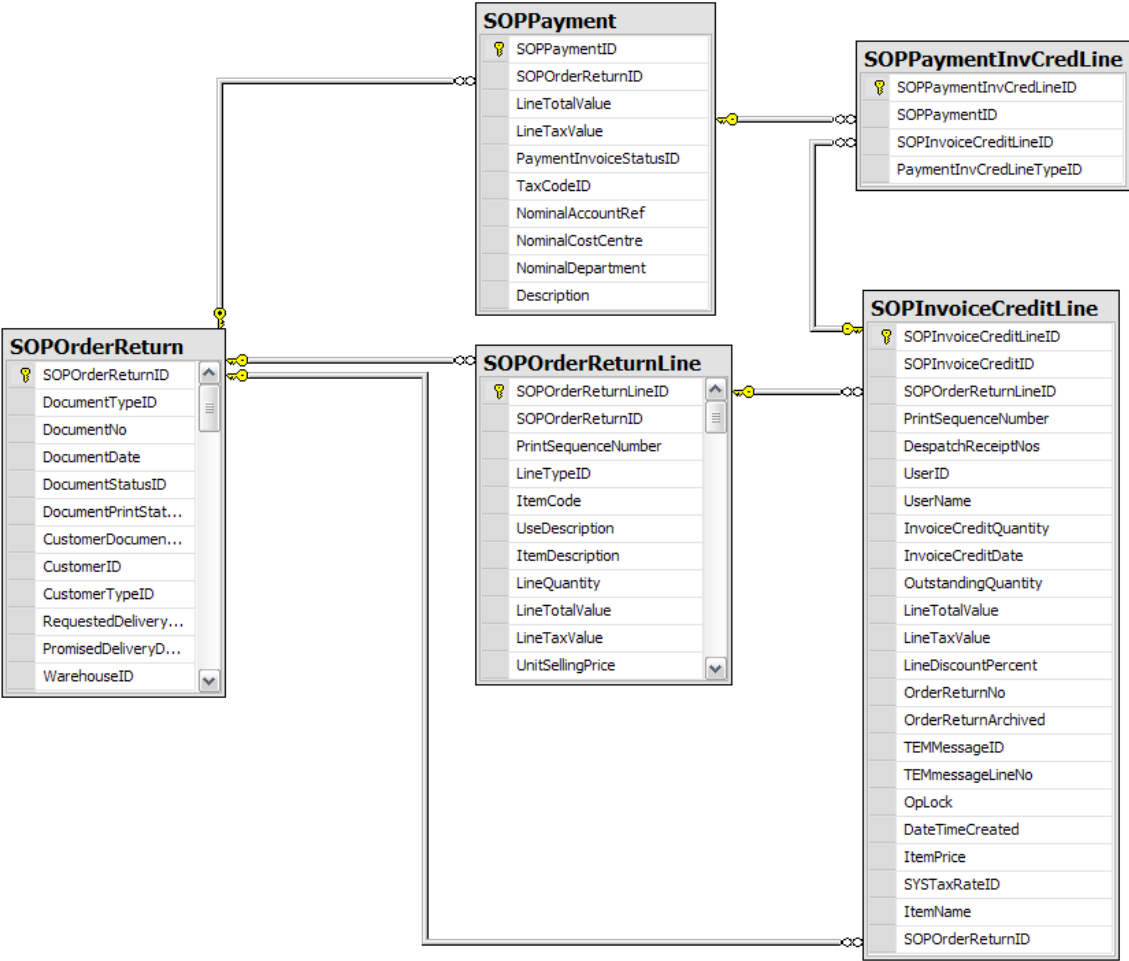
SOP Payments are recorded and held in the SOPPayment and SOPPaymentInvCredLine tables.

For each SOP Payment a record is created in the SOPPayment table which is linked to the SOP Order via the SOPOrderReturn table.

For each SOP Payment one or more records are also created in the SOPPaymentInvCredLine table.

These are all linked to the one SOPPayment record and each SOPPaymentInvCredLine record is linked to its corresponding SOPInvoiceCreditLine record.

**Note:** Sales orders migrated from Line 50 or Line 100 will not have SOPPayment records. These links cannot be created during the data conversion.



## SOP Quotations

A SOP Quotation is held in the SOPOrderReturn (and SOPOrderReturnLine) table just like an order, but it has a SOPOrderReturn.DocumentTypeId of 3.

The DocumentNo will be the quotation number.

When the quotation is converted into a Sales order, a new SOPOrderReturn (and SOPOrderReturnLine) records are created where the DocumentNo is the actual Sales order number and the DocumentTypeId will be 0.

These two SOPOrderReturn records are not linked in the database but can be connected to each other by the SourceDocumentNo linking to the DocumentNo.

## SOP Pro Formas

This works exactly the same as a quotation except the SOPOrderReturn.DocumentTypeId for the pro forma is 2 and the DocumentNo is the pro-forma number.

When the pro-forma is converted into a sales order, new records are created exactly as for a quotation.

## SOP Repeat Orders

Again the repeat order details are held in the SOPOrderReturn (and SOPOrderReturnLine) records but this time the SOPOrderReturn.DocumentTypeId is 4 and the DocumentNo is the repeat order number.

There will also be a SOPRepeatOrderSchedule record linked to the SOPOrderReturn on SOPOrderReturnId. This holds the details of how often the order is to be repeated and how many times it has been used to generate a sales order etc.

Each time a sales order is generated from the repeat order a new SOPOrderReturn (and SOPOrderReturnLine) records are created exactly as described for the quotation, linking back to the original repeat order in the same way.

**Note:** For all these special types of SOPOrderReturn, there may not be a linked sales order. They will only exist if one has been generated from the source document.



# Purchase Order Processing

## POP Orders

When an order is taken through the application the following records are set up:

- POPOrderReturn is the order 'header' which contains information relating to the order as a whole, but not to individual order lines.
- POPOrderReturnLine records contain the details of each order line of the various [types](#).

The basic order will consist of **one** POPOrderReturn, linked to **one or many** POPOrderReturnLine records on [POPOrderReturnId](#). The order number is held in POPOrderReturn.[DocumentNo](#).

The POPOrderReturn will link to **one** PLSupplierAccount record POPOrderReturn.[SupplierId](#) links to PLSupplierAccount.[PLSupplierAccountId](#). The order header may link to **one or many** POPCancelledLine records on [POPOrderReturnId](#).

For each POPOrderReturnLine where the [LineTypeId](#) is 0 (Standard) there will be **one** POPStandardItemLink record linked on [POPOrderReturnLineId](#). This links to the StockItem on [ItemId](#).

The POPOrderReturn may link to **one** POPDocDelAddress on [POPOrderReturnId](#) record holding the delivery address for the main order. However each POPOrderReturnLine may also link to **one** POPOrdReturnLineDelAddress on [POPOrdReturnLineDelAddressID](#), if certain order lines are for direct delivery.

**Note:** One POPOrdReturnLineDelAddress may link to **many** POPOrderReturnLines in the same order.

Any order line which has stock still on order but not received (i.e. the [LineTypeId](#) is 0 and the associated StockItem.[StockItemTypeId](#) is 0 (Stock) and the [DespatchReceiptQuantity](#) is less than the [LineQuantity](#)), will have **one** associated POPOnOrderLine record linked on [POPOrderReturnLineId](#). This, in turn, will be linked to **one** POPOnOrderBalance record on [POPOnOrderBalanceId](#). This provides the link through to the **one** WarehouseItem record which holds the total [QuantityOnPopOrder](#). This field is displayed on the [Stock > Enquiries > Stock Status](#) workspace in the [Warehouse](#) Details tab..

**Note:** The WarehouseItem may link to **many** POPOnOrderBalance records.

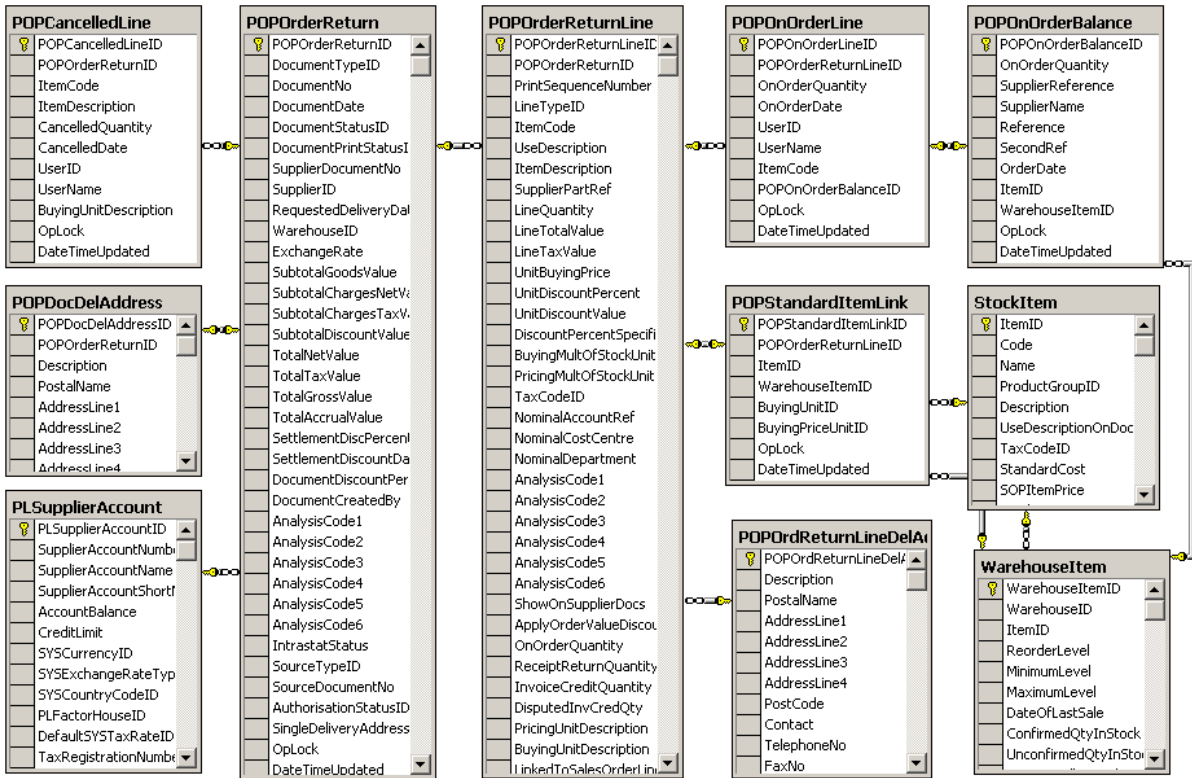


Diagram: Purchase Orders

## POP Receipts

To check the data relating to POP receipts, viewable using the [Receipt/Despatch Lines](#) tab in the [Purchase Document Lines](#) workspace:

If any part of an order has been despatched then these records must exist.

- Find the POPOrderReturn record using the DocumentNo (the order number displayed on screen).
- Then link the POPOrderReturn record to its **one or more** POPOrderReturnLine using the [POPOrderReturnId](#).

### For All Items

- The POPOrderReturnLine should link to one or more POPReceiptReturnLine records - on POPOrderReturnLineID – where the total of the POPReceiptReturnLine.ReceiptReturnQuantity fields is equal to the POPOrderReturnLine.ReceiptReturnQuantity.
- Each POPReceiptReturnLine will link to one POPReceiptReturn record linked via POPReceiptReturnId.

**Note:** One POPReceiptReturn may link to **one or more** POPReceiptReturnLines not necessarily all for the same order.

### For Traceable Items

- Each POPReceiptReturnLine will link to one or more TracePOPRcptRtnLine record on POPReceiptReturnLineId where the total of TracePOPRcptRtnLine.Quantity matches the POPReceiptReturnLine.ReceiptReturnQuantity.
- Each TracePOPRcptRtnLine will link to one TraceableBinItem on TraceableBinItemId.
- The TraceableBinItem will then link to one TraceableItem via TraceableItemId; this will give the specific batch or serial number for this POP receipt in the TraceableItem.IdentificationNo field.

When viewing the deliveries for an order line in the application, if you select the [Traceable Receipt/Despatch Details](#) tab it is these traceable records that are used to obtain the information displayed.

Likewise selecting a specific batch or serial in [Stock > Traceable Items](#) and select the [Receipts](#) tab – these traceable records are used to link through to the Purchase order.

**Note:** One TraceableBinItem may link to one or more TracePOPRcptRtnLine records (if it is a batch rather than a serial item). One TraceableItem may link to one or more TraceableBinItem records (if it is a batch rather than a serial item).  
Purchase orders migrated from Line 50 or Line 100 will not have TracePOPRcptRtnLine records – as these links are impossible to create during data conversion.

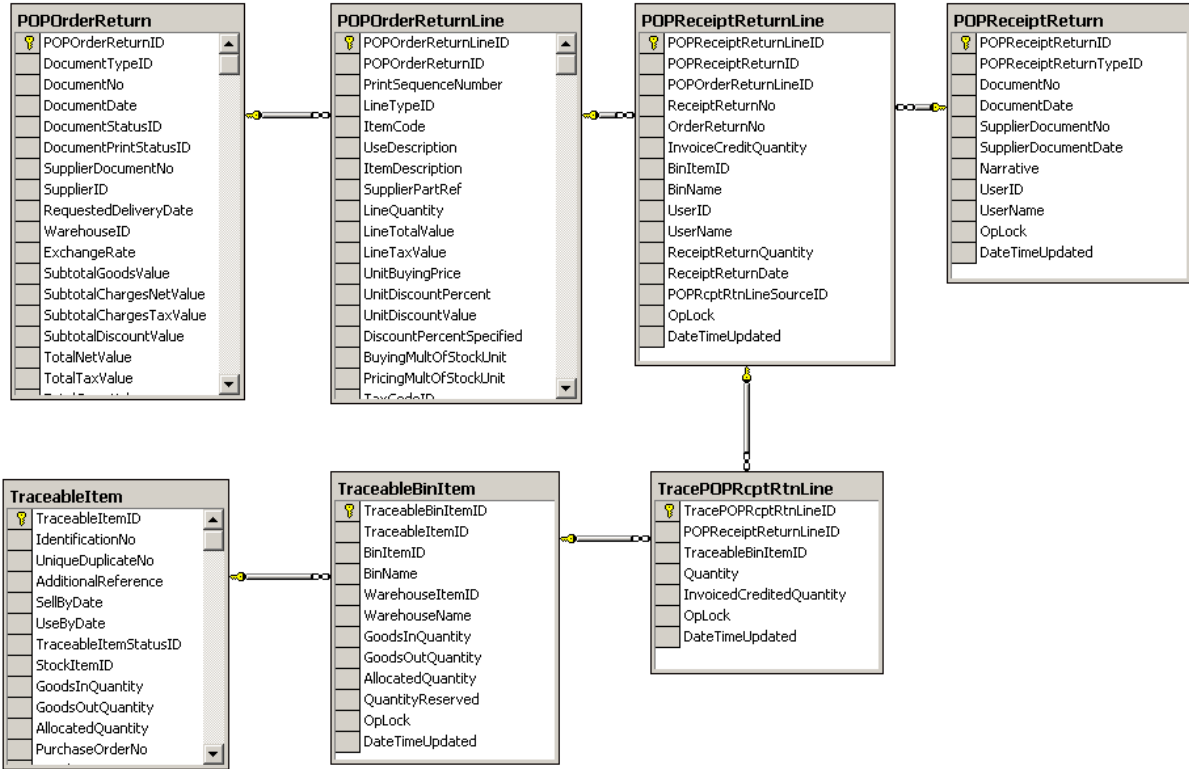


Diagram: Purchase Receipts

## POP Invoices

To check the data relating to POP invoices, viewable using the **Invoice/Credit Lines** tab in the **Sales Document Lines** workspace:

If any part of an order has been invoiced then these records must exist.

- Find the POPOrderReturn record using the DocumentNo (the order number displayed on screen).
- Then link the POPOrderReturn record to its **one or more** POPOrderReturnLine records using the POPOrderReturnId.

If the order line has a non-zero quantity in **InvoiceCreditQuantity**, it must have additional linked records as follows:

### For All Items

- The POPOrderReturnLine should link to **one or more** POPInvoiceCreditLine records - on POPOrderReturnLineID – where the total of the POPInvoiceCreditLine.InvoiceCreditQuantity fields is equal to the POPOrderReturnLine.InvoiceCreditQuantity.

**Note:** There will only be a linking invoice ‘header’ record (the POPInvCredDispute record - linked on POPInvCredDisputeId), if the invoice is in dispute. Otherwise, no invoice header record will exist for the invoice lines.

### For Traceable Items

- Each POPInvoiceCreditLine will link to **one or more** TracePOPInvCredLine record on POPInvoiceCreditLineId where the total of the TracePOPInvCredLine.Quantity matches the POPInvoiceCreditLine.InvoiceCreditQuantity.
- Each TracePOPInvCredLine will link to **one** TraceableBinItem on TraceableBinItemId.
- The TraceableBinItem will then link to **one** TraceableItem via TraceableItemId; this will give the specific batch or serial number for this POP invoice line in the TraceableItem.IdentificationNo field.

**Note:** One TraceableBinItem may link to one or more TracePOPInvCredLine records (if it is a batch rather than a serial item). One TraceableItem may link to one or more TraceableBinItem records (if it is a batch rather than a serial item).

Purchase orders migrated from Line 50 or Line 100 will not have TracePOPInvCredLine records as these links were impossible to create during data conversion.

### POP Order Lines that have both Receipts and Invoices

If an order line has a receipt and the invoice for that receipt has also been received, there is an additional table the POPRcptRetInvCredLine that links to **one** POPReceiptReturnLine on POPReceiptReturnLineId and **one** POPInvoiceCreditLine on POPInvoiceCreditLineId.

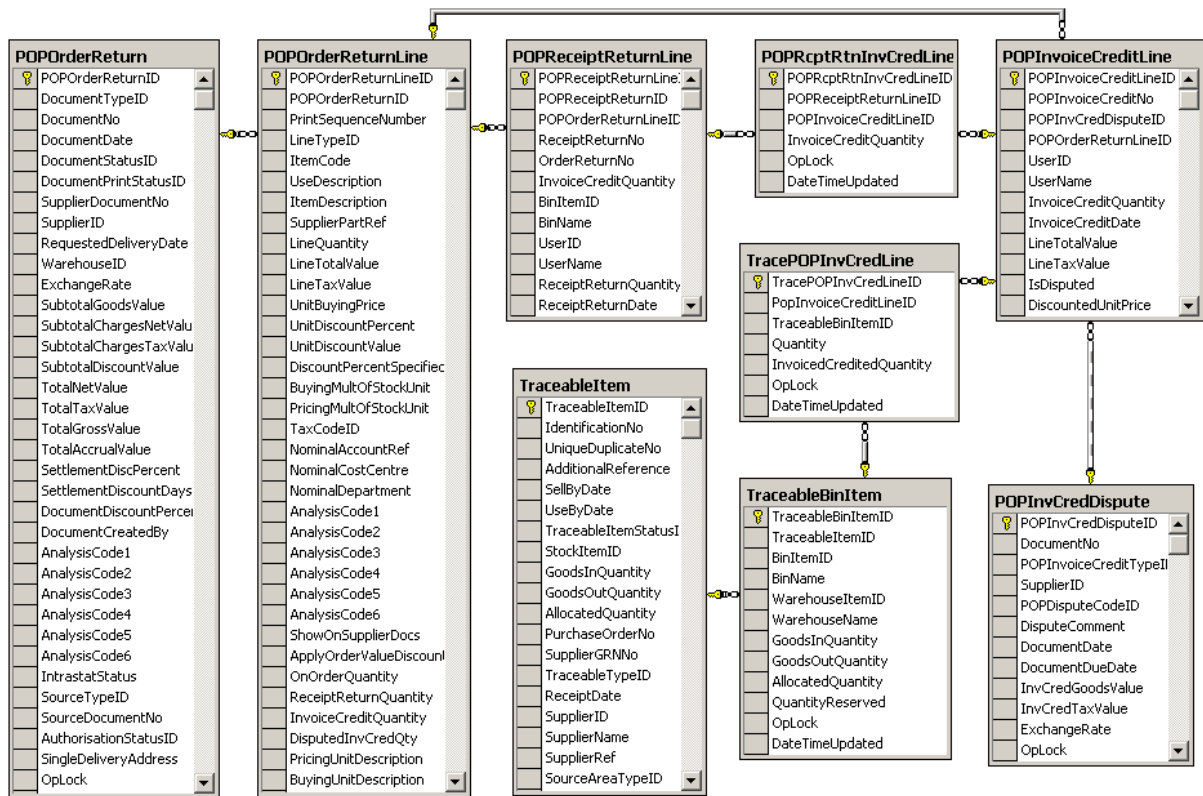


Diagram: Purchase Invoices

# Stock

## Item Details

- The basic stock item record is StockItem which is linked to **one** ProductGroup on **ProductGroupId** (a ProductGroup can be linked to **many** StockItem records).

**Note:** The default settings on the ProductGroup can often be overridden at the Item level.

- The StockItem may be linked with many StockItemSupplier records, on **ItemId**, each of which will link to one PLSupplierAccount record in the Financials, on **SupplierId** (links to PLSupplierAccountId).

There should usually be one of these StockItemSupplier records with the **Preferred** flag set to True (Value 1).

These can be viewed through the application on the **Suppliers** tab of the **Stock Status** workspace

- The StockItem will also link to **four** StockItemNominalCode records, on **ItemId**, that in turn link to one NLNominalAccount record on **NominalCodeId** (links to NLNominalAccountId).

The StockItemNominalCode.**NominalUsageld** will tell you which code this is for (Stock, Issues, Revenue, and Accrued Receipts).

Viewable on the **Analysis** tab of the **Stock Status** workspace

- The ProductGroup similarly will link to **four** ProdGroupNominalCode records, on **ProductGroupId**; each will link to one NLNominalAccount record as stock item above.

The ProdGroupNominalCode.**NominalUsageld** will indicate which code this is for.

- If there is an alternative item specified this will be recorded in the **one or more** AlternativeItem records linked to the Stock item on **ItemId**.

These can be viewed through the application on the **Alternatives** tab of the **Stock Status** workspace

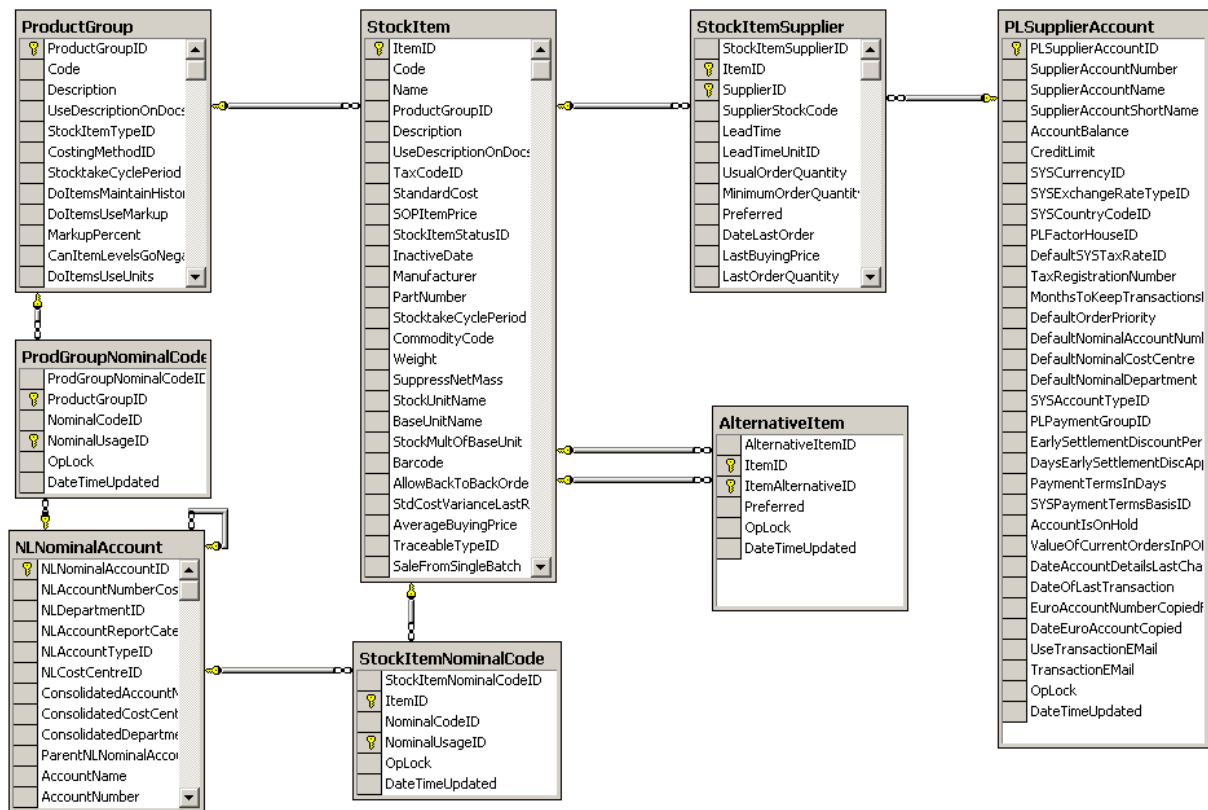


Diagram: Stock Items



## Search Categories

This is another area where a many-to-many relationship has to be avoided by the use of linking tables.

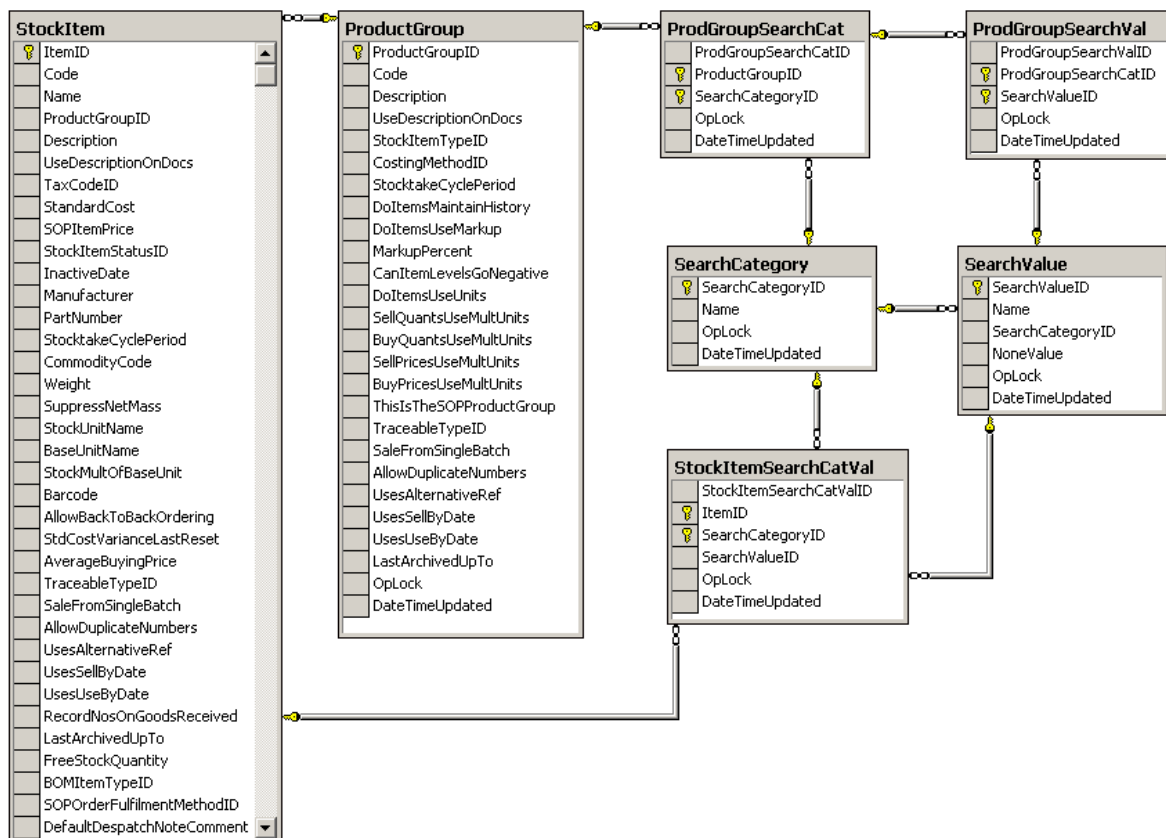
- Each item can have many search categories and a search category can be applicable to many items. It is the same with search values.

To avoid the **many-to-many** relationship, a table called the StockItemSearchCatVal is used. This contains no information itself but is used to link to **one** StockItem on **ItemId** and **one** each of SearchCategory, on **SearchCategoryID**, and SearchValue, on **SearchValueID**.

- A product group could have many search categories and each category could have many search values. The search category can apply to many product groups and search values may apply to many search categories.

So the ProductGroup links to **one or more** ProductGroupSearchCat records, on **ProductGroupID** which links to **one** SearchCategory on **SearchCategoryID**. Each ProductGroupSearchCat links to **one or more** ProductGroupSearchVal, on **ProductGroupSearchCatID**, which link to **one** SearchValue on **SearchValueID**.

- These categories are viewable for a stock item on the **Search Categories** tab of the **Stock Status Enquiry** workspace. To find stock items using a particular category or category/value combination use **Enquiries > Stock item Search**



## Stock Locations

Even if multiple locations is turned off in Stock Settings, there will still be one Warehouse called 'Home' with one Bin called 'Unspecified'. These are viewable for a stock item through the [Stock Status](#) workspace on the [Warehouse Details](#) and [Bin Details](#) tabs.

These records will exist:

- There is **one** Warehouse record for each 'Warehouse' set up on the system.
- For every stock item that is linked to that warehouse there will be **one** WarehouseItem record. One Warehouse will therefore link to **one or many** WarehouseItem records, on [WarehouseId](#).
- For each WarehouseItem record there will always be **one or many** BinItem records linked on [WarehouseItemId](#).

So for every stock item at a warehouse there will at least be **one** BinItem record for the 'unspecified' bin, but there could be **many** BinItem records. Both the WarehouseItem and BinItem link to the StockItem on [ItemId](#).

**Note:** There is no Bin record in the way that there is a Warehouse record. A bin is only deemed to exist if there are stock item(s) associated with it, it has no existence if there are no associated stock items. Therefore only BinItem records are required.

When using [Stock > Stock records > Stock Locations](#), you can select a stock item and delete a bin (which will delete the BinItem) and a warehouse (which will delete the WarehouseItem) but this will not delete the Warehouse record itself.

Warehouse records can only be deleted by using [Stock Maintenance > Locations](#).

**Note:** The configurable names option means that these entities may be referred to as something other than 'warehouse' and 'bin' in the application, but the underlying database will always use the standard names.

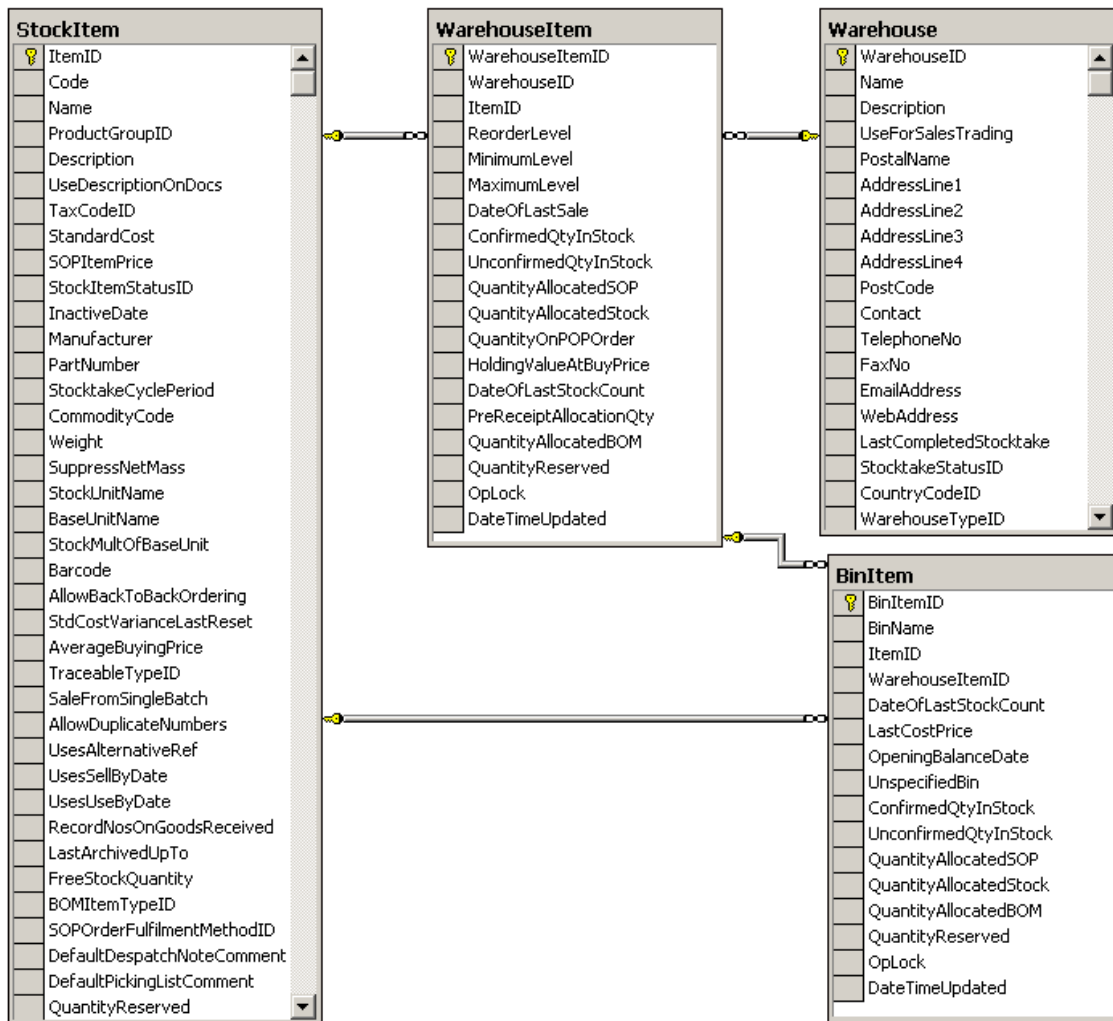


Diagram: Stock Items

## Units of Measure

This is an area of the data that is made complex by the need to break some of the many to many relationships between the entities, which are not acceptable in relational database design.

For example, **one** Stock item can have **many** Units of Measure, and **one** Unit of Measure can be applied to **many** Stock items; this is a **many to many** relationship.

To break this, we need to use a linking table that has a **one-to-many** relationship with each of the above tables. This linking table is the StockItemUnit table.

**One** StockItem can have **many** StockItemUnits but each StockItemUnit can only apply to **one** StockItem.

Likewise **one** Unit can relate to **many** StockItemUnits but each StockItemUnit can only relate to **one** Unit.

These are viewable for a stock item through the **Stock Status** workspace on the **Units of Measure** tab

**Note:** Even if an item does not use different units of measure there will still be an 'Each' unit attached to it so the following applies to all stock items.

- The units that apply to an item are held in the StockItemUnit table; each StockItem will link to **one or more** by **ItemId**. There will always be at least one of these records for each item.
- Each StockItemUnit will link to **one or more** StockItemUnitUOMType on **StockItemUnitId**; these will have the **UnitOfMeasureType** of 1- Stock, 2 – Base , or 3 – Additional.

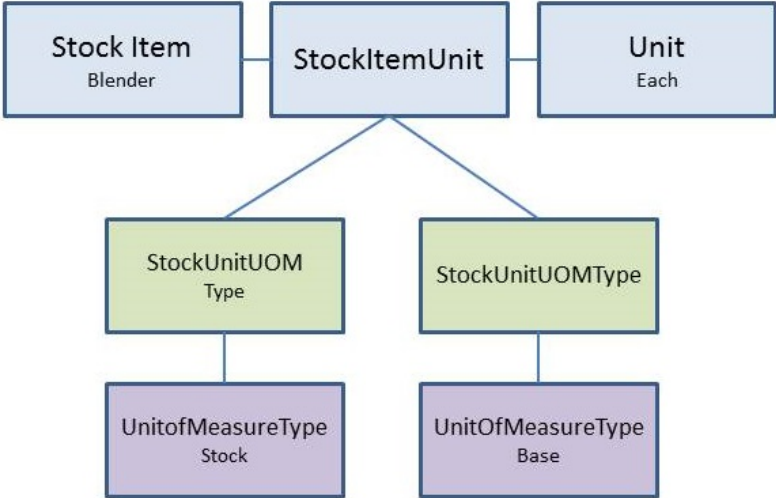
Items that do not use multiple units will have two of these records, one for Stock and one for Base. The Unit that the StockItemUnit links to (on **UnitId**) gives the name of that particular Unit (e.g. EACH).

Stock Items that use multiple units will have more than one StockItemUnit record linking to the different Unit records (to get the name of the unit). Each of these will link to **one or more** StockItemUnitUOMType records to show what type of Unit this is for this item.

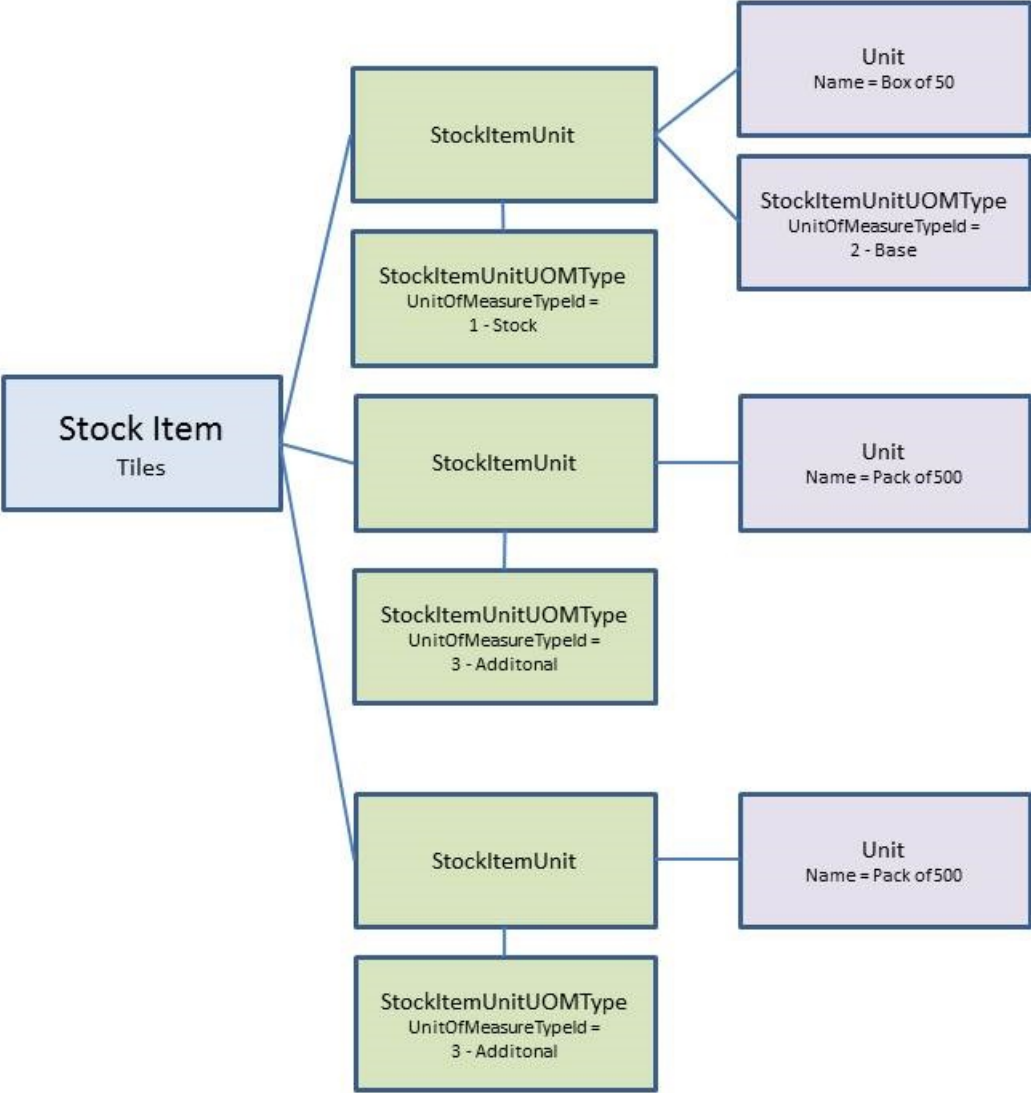
**Note:** The StockItemUnit record holds no meaningful data itself. It is there to break the many-to-many relationship that exists between the StockItem and the Unit tables. Likewise, the StockItemUnitUOMType record holds no meaningful data itself. It is there to break the many-to-many relationship between the Unit and UnitOfMeasureType tables.

**Examples:**

*Stock Item - Blender* does not use multiple units but uses 'Each' for both Stock and Base units. The records would be:



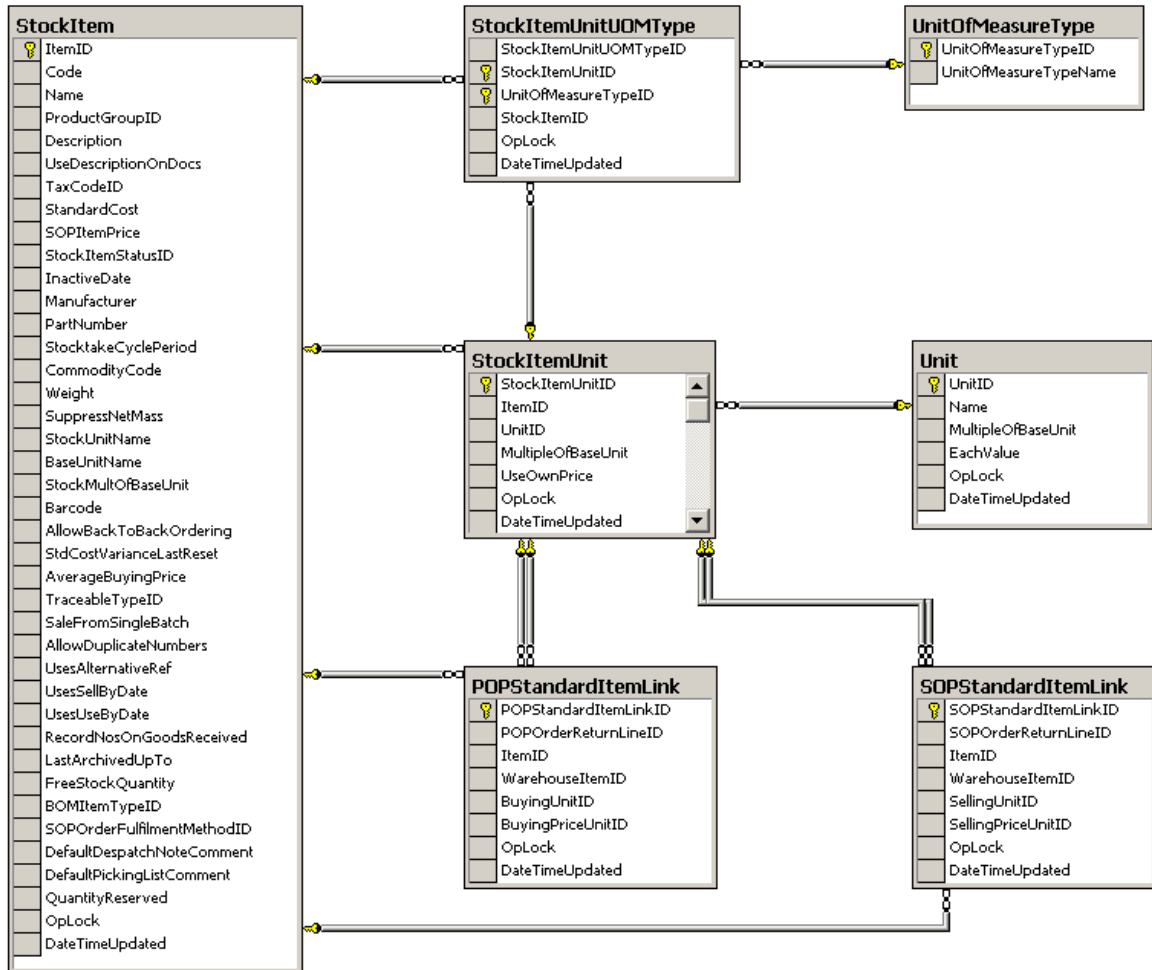
Stock Item - Tiles is stocked in Box of 50 and uses Box of 50 as the Base unit, but is bought in Pack of 500 and sold in Boxes of 50 and 100.



When connecting a stock item to a Sales Order line, the link is always made between the SOPStandardItemLink record and one **StockItem**.

In addition a link is also made to **one or more** StockItemUnit records, on **StockItemUnitId**, that relate to the SellingUnit and SellingPriceUnit for this item. This could be the same StockItemUnit or different.

It is the same with the link to a Purchase Order line. The link is via the POPStandardItemLink record that links to **one** Stockitem and also **one or more** StockItemUnit records that relate to the BuyingUnit and BuyingPriceUnit. (Again, this could be the same StockItemUnit or not.)



## Stock levels

Even if the site does not use multiple locations there will still be one Warehouse called 'Home' and one Bin called 'Unspecified'. So the following will be the same regardless of the use of locations.

### To look at total stock

Start by finding the StockItem using the Code field.

- This will be linked to **one or more** BinItem records on **ItemId**.
- The Bin item is linked to **one** WarehouseItem on **WarehouseItemId**; this links to the Warehouse on **WarehouseId** which tells us which warehouse the bin belongs to.

**Note:** The WarehouseItem may link to **one or more** BinItem records.

- Using the BinItem.**BinItemId**, find all linking MovementBalance records (the actual 'bits' of stock).
- The total stock is calculated by subtracting the MovementBalance.**StockLevelIssued** from the MovementBalance.**OpeningStockLevel** on each MovementBalance.
- The total of these will give you the total stock for this stock item in this bin.
- To find the total stock at a warehouse, total all MovementBalances for all BinItems linked to the same WarehouseItem (on **WarehouseItemId**).
- To find total stock anywhere, just total all the MovementBalance records for this stock item i.e. linked to the same StockItem using **ItemId**.

In the application if you use **Stock > Enquiries > Stock Status** and then select the **Receipts and Prices** tab, you will see details of each MovementBalance showing the quantity and buying price.

If the Buying price is zero, this may be stock that has been received from a Supplier, but where the invoice has not yet been received, therefore the actual price of these goods is not yet known.

### To find Free stock

It is necessary to consider the AllocationBalance records that are linked to the BinItem.

**Note:** Allocations are at bin level; they do not relate directly to the 'bits' of stock (MovementBalance records).

- For the free stock at a bin, total all the AllocationBalance.AllocatedQuantity fields that link to the same bin via **BinItemId** and subtract this from the total stock quantity at that bin, as calculated above
- For the free stock quantity at a warehouse, total all the free stock quantities at the bins linked to the same WarehouseItem via **WarehouseItemId**.



The system itself keeps some running totals:

- BinItem.ConfirmedQuantityInStock is the total of linked MovementBalance records with MovementBalanceType of 0 – Confirmed and 2 – Shortfall.  
  
(A shortfall is a MovementBalance record that has more StockLevelIssued than OpeningStockLevel and relates to negative stock situations.)
- BinItem.UnconfirmedQuantityInStock is the total of linked MovementBalance records that have a MovementBalanceType of 1 – Not Confirmed and 4 – Unconfirmed Return.  
  
These types relate to items that have been received on a POP Order or returned on a POP Return but where the Invoice or Credit Note has not yet been entered into the system).
- BinItem.QuantityAllocatedSOP and BinItem.QuantityAllocatedStock.  
  
These are the totals of the linked AllocationBalance .AllocatedQuantity fields.  
  
The ones that have a linking SOPAllocation (linked on AllocationId) will be in QuantityAllocatedSOP.  
  
The ones that have a linking BOMAllocBalance will be in QuantityAllocatedBOM  
  
Any others will be in QuantityAllocatedStock.
- So to get the free stock at a bin the formula is BinItem. (ConfirmedQuantityInStock + UnconfirmedQuantityInStock) - (QuantityAllocatedSOP + QuantityAllocatedBOM + QuantityAllocatedStock).  
  
WarehouseItem.ConfirmedQuantityInStock and UnconfirmedQuantityInStock are the totals of the same fields in all linked BinItem records.
- Likewise with free stock you need to calculate for the WarehouseItem. (ConfirmedQuantityInStock + UnconfirmedQuantityInStock) - (QuantityAllocatedSOP + QuantityAllocatedBOM + QuantityAllocatedStock).
- StockItem.FreeStockQuantity is the total of all free stock quantities at all the warehouses.

**Note:** Every time a stock item is affected by the system, these totals have to be amended.



Diagram: Stock Levels

## Traceable Items

- Any item that is traceable will have **one or more** TraceableItem records linking to the Stockitem on **StockItemId** (links to StockItem.ItemId). These are viewable using **Stock > Enquiries > Traceable Items**

If the item is serial numbered there will be one of these records for each serial number with the serial number in **IdentificationNumber**.

For batch numbered items there will be a TraceableItem for each batch number associated with the item, the batch number held in **IdentificationNumber**.

- This record will link to **one or more** TraceableBinItem records on **TraceableItemId**.  
These records will continue to exist even when the item(s) have been completely sold.
- Each TraceableBinItem in turn may link to **one or more** TraceableMovementBal records on **TraceableBinItemId**, but only if the costing method for this item is Actual Cost.

## Goods In Stock

- To find which traceable items are in stock currently, subtract the TraceableItem.**GoodsOutQuantity** from the **GoodsInQuantity**.

If this leaves a positive number then there are some in stock (if it's a serial item then this particular one is in stock).

If the balance is 0 then none of this batch (nor this individual serial item) is in stock.

- Free stock is calculated by subtracting the **AllocatedQuantity** from the stock quantity obtained above.
- The TraceableBinItem links to the BinItem on **BinItemId**.  
(One BinItem may link to **one or more** TraceableBinItems).
- The BinItem then links to the MovementBalance records, but only if there is actually any of this particular serial/batch in stock in that bin. If there is none in stock then a linking MovementBalance record will not exist.

In the application, using **Stock > Enquiries > Traceable Items**, select the item and select the **Locations** tab and details of where the item is held (if any are in stock) will be shown.

## Goods Received from Suppliers

- To find where and when the traceable item was purchased, you can link the TraceableBinItem to the Purchase order (if the item was purchased via POP).

Use the **one or more** linking TracePOPRcptRtnLine linked on **TraceableBinItemId** (only more than one if it is a batch).

This links to **one** POPReceiptReturnLine record on **POPReceiptReturnLineId** which links to **one** POPOrderReturnLine on **POPOrderReturnLineId**, which links to **one** POPOrderReturn on **POPOrderReturnId** which gives you the POP order number in **DocumentNo**, and order date in **DocumentDate**.

In the application using **Stock > Enquiries > Traceable Items** select the item and select the **Receipts** tab and details of the associated POP orders will be shown.

### Goods despatched to Customers

- If the item(s) have been despatched on Sales Order(s) then there will be **one or more** linking TraceSOPDespRcptLine records linked on **TraceableBinItemId** (only more than one if it is a batch).

These records link to **one** SOPDespatchReceiptLine records on **SOPDespatchReceiptLineId**, which links to **one** SOPOrderReturnLine on **SOPOrderReturnLineId**, which links to **one** SOPOrderReturn record via **SOPOrderReturnId** which gives the Sales order number in **DocumentNo**.

In the application using **Stock > Enquiries > Traceable Items** select the item and select the **Issues** tab and details of the associated SOP orders will be shown.

**Note:** Orders migrated from Line 50 or Line 100 will not show the details of SOP/POP orders as the linking TracePOPrcptRtnLine and TraceSOPDespRcptLine records do not exist. It is impossible to create these during data migration.

### Goods allocated

- To see allocations for a TraceableItem link the TraceableBinItem to **one or more** TraceableAllocationNoteal records on **TraceableBinItemId**.

Each of these must directly link to **one** AllocationBalance record on **AllocationBalance Id** (links to AllocationBalance **AllocationId**).

Thus the allocated stock and batch/serial numbers must always be in sync.

- To find out which sales order these items are allocated to, (if they are SOP allocations), the AllocationBalance record will link to **one or more** SOPAllocationLine on **AllocationId**.

This will link to **one** SOPOrderReturnLine on **SOPOrderReturnLineId** which links to **one** SOPOrderReturn on **SOPOrderReturnId** which contains the sales order number in **DocumentNo**.

In the application, using **Stock > Enquiries > Traceable Items** select the item and select the **Allocations** tab and details of the associated SOP orders will be shown.

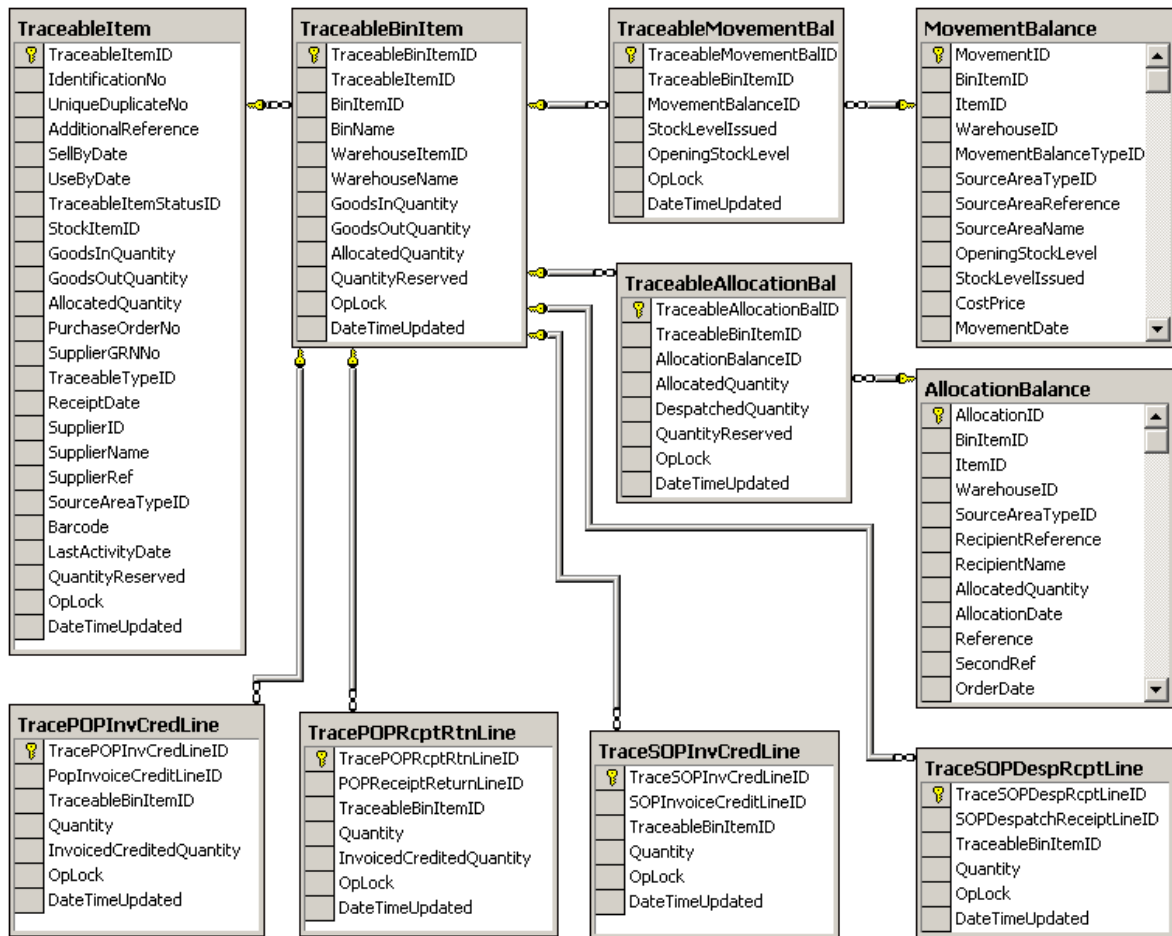


Diagram: Stock Traceability

# Price Book

## Prices

- There must always be at least one PriceBand record, with [ThisIsTheStandardBand](#) set to 1 (true).  
There may be additional PriceBand records; if so, these must have [ThisIsTheStandardBand](#) set to 0 (false).
- Each PriceBand record must link to **many** StockItemPrice records, on [PriceBandId](#), one for every StockItem. These link to the StockItem on [ItemId](#). These are viewable for a stock item using [Price Book > Enquiries > Stock item Prices](#).
- Each customer record, SLCustomerAccount, is linked to a particular PriceBand by SLCustomerAccount.PriceBandId linking to PriceBand.PriceBandId. Thus all customers must be linked to one price band but only one. These are viewable for a customer using [Price Book > Enquiries > Customer Prices](#).

When a new customer record is set up it will by default link to the standard price band.



## Discounts

Discounts can be set up for both stock items and product groups.

- To be used within the system, the discounts must be linked with a CustomerDiscountGroup that will be linked to **one or many** customers (SLCustomerAccount).
- To look at the discounts a customer has available to them, link the SLCustomerAccount record to **one** CustomerDiscountGroup using the **CustomerDiscountGroupId**.

A customer can only be linked to one group.

However **one** discount group can have **many** discounts and **one** discount could relate to **many** discount groups. This is a **many-to-many** relationship that must be broken down.

- These are viewable for a customer using Price Book > Enquiries > Customer Discounts

### Product group discounts

- For discounts relating to product groups, link the CustomerDiscountGroup to **one or many** CustProdGroupDiscount records using **CustomerDiscountGroupId**.

Then link each of these to **one** ProductGroupDiscount using the **ProductGroupId**.

This will link to **one** ProductGroup using the **ProductGroupId**.

**Note:** Product Group discounts can only be of the **type** 'percentage'.

- Each ProductGroupDiscount must link to **one or more** ProdGroupQtyDiscBreak records on **ProductGroupId**.

These contain the details of the quantity breaks and percentage value associated with each break.

- These are viewable for a stock item using Price Book > Enquiries > Stock item Discounts

### Stock item discounts

- For discounts relating to stock items, link the CustomerDiscountGroup to **one or many** CustStockItemDiscount record using **CustomerDiscountGroupId**.

Each of these will link to **one** StockItemDiscount using **StockItemDiscountId**.

This will link to **one** StockItem using **StockItemId** (links to StockItem.ItemId).

- Stock item discounts can be one of 3 **types** but all must have **one or many** StockItemQtyDiscBreak records linked on **StockItemDiscountId**.

These contain details of any quantity breaks and the percentage, amount or fixed price associated with each break.

- These are viewable for a stock item using Price Book > Enquiries > Stock item Discounts

**Note:** You should not link the customer discount group to both product group discounts and stock item discounts in the same SQL query; otherwise the results will be meaningless. You must link each discount type individually.

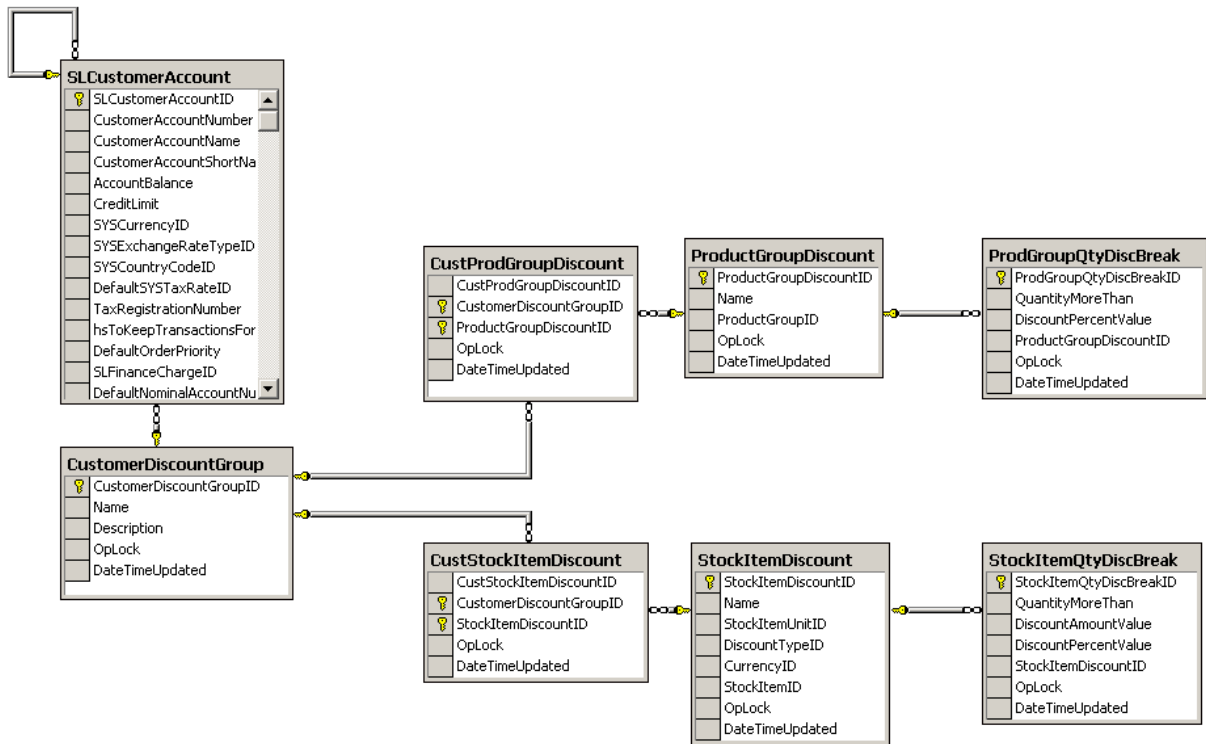


Diagram: Customer Discount Structures



# Bill of Materials module

## BOMs

These tables together define the 'recipes' for your bills of material:-

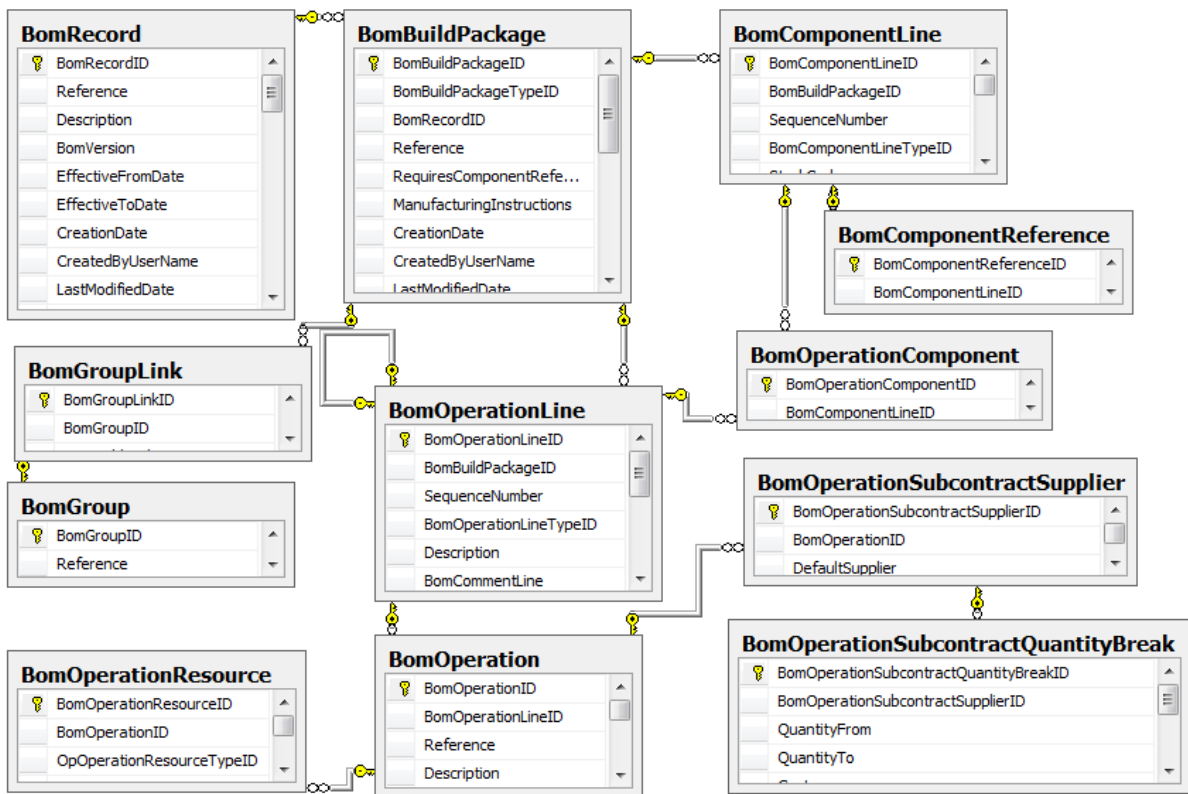
The BomRecord is the header for each version of a BOM. This has no physical referential links to the StockItem, but does contain the StockItem.Code in the **Reference** field and the StockItem.Name in the **Description** field.

This links to **two** BomBuildPackage records on **BomRecordId**. Only the record that has the **BomBuildPackageTypeId** of 0 (meaning Build) is relevant however. Type 1 - Salvage is for possible future requirements. The BomBuildPackage may link to **one or more** BomGroupLink records each of which links to **one** BomGroup. This is to allow a Bom to belong to many groups.

The BomBuildPackage then links to **one or many** BomComponentLine records. The **BomComponentLineTypeId** tells you what sort of line this is – component, subassembly etc. If this is a component or a subassembly, the **StockCode** field will contain the StockItem.Code although again there is no referential link in the database.

The BomBuildPackage may also link to **none, one or many** BomOperationLine records. The **BomOperationLineTypeId** tells you what sort of line this is. If this is type 0 or 1 then this record will link to **one** BomOperation, which in turn may link to **none, one or many** BomOperationResource records.

The BomOperation may link to **one or many** BomOperationSubcontractSupplier records which in turn may link to **one or many** BomOperationSubcontractQuantitybreak records.



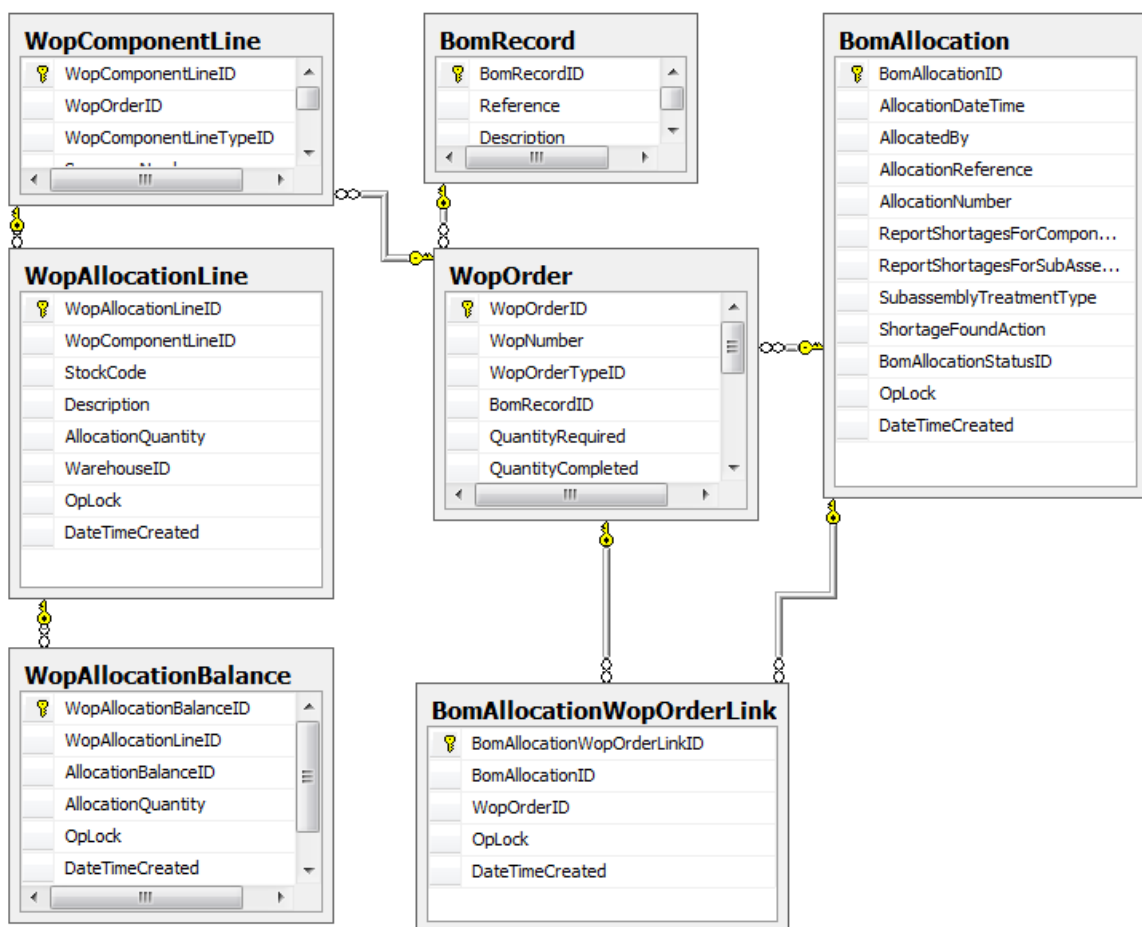
## BOM Allocations

The BomAllocation table provides the header record for a whole allocation session. This will link to **one or many** WPOOrder records, one for each different finished item entered.

For example; If you allocate to build 1 Arizona kitchen and 2 New England kitchens, you will get one WopOrder for quantity 1 of the Arizona kitchen and one WopOrder for quantity 2 of the New England kitchen.

The WopOrder links back to **one** BomRecord for the finished item. It also links to **one or many** WopComponentLine records, one for each different component or sub-assembly included in the allocation. These link to **one or many** WopAllocationLine records; there will be multiple if the allocated stock comes from different warehouses.

The WopAllocationLine will link to **one or many** WopAllocationBalance records. The WopAllocationBalance.**AllocationBalanceId** will contain the Id of the AllocationBalance within the stock system (which links to the StockItem), however there is no referential link to this.



## BOM Builds

These records are a history of all builds of finished items, which have been carried out using the Sage 200 BOM module.

These hold similar information to the BOM Allocations records, but allocation details are deleted once the build process has been completed. The build records therefore keep this information for historical and reporting purposes.

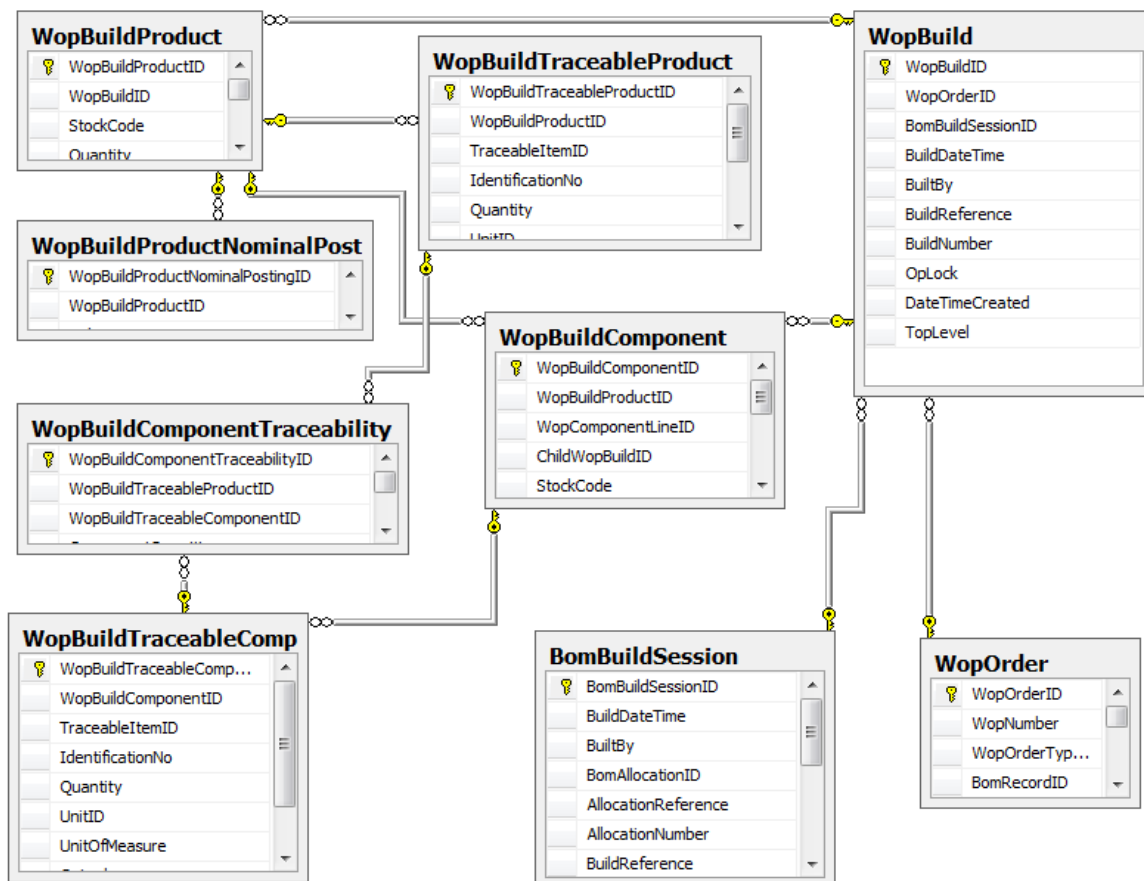
The BomBuildSession is the header record for the whole build session. This will link to **one** WopBuild record. This will link back to **one** WopOrder.

**Note:** The WopOrder may link to **many** WopBuild records if the required quantity of a finished item was built in separate build sessions.

The WopBuild links to one or more WopBuildProducts, one for each separate finished item that was built in the session. If the finished item was traceable this will link to **one or more** WopBuildTraceableProduct, one for each serial or batch number built.

The WopBuild also links to **one or more** WopBuildComponents, one for each separate component or subassembly used in the build session. If the component was traceable the WopBuildComponent will link to **one or more** WopBuildTraceableComponents, one for each serial or batch number used.

To facilitate full traceability from the finished product back to the components used to build it, the WopBuildComponentTraceability record links **one** WopBuildTraceableProduct to **one** WopBuildTraceableComponent, thus allowing the **many-to-many** relationship between these tables.



# Project Accounting

## Projects

In the application this data can be seen through [Project Accounting > Project Enquiries > Projects > Projects](#).

A PCProjectItem is created for all elements in a project structure: Projects, Groups and Project Items. The PCProjectItemTypeld points to the [look-up](#) table that denotes the type of project item.

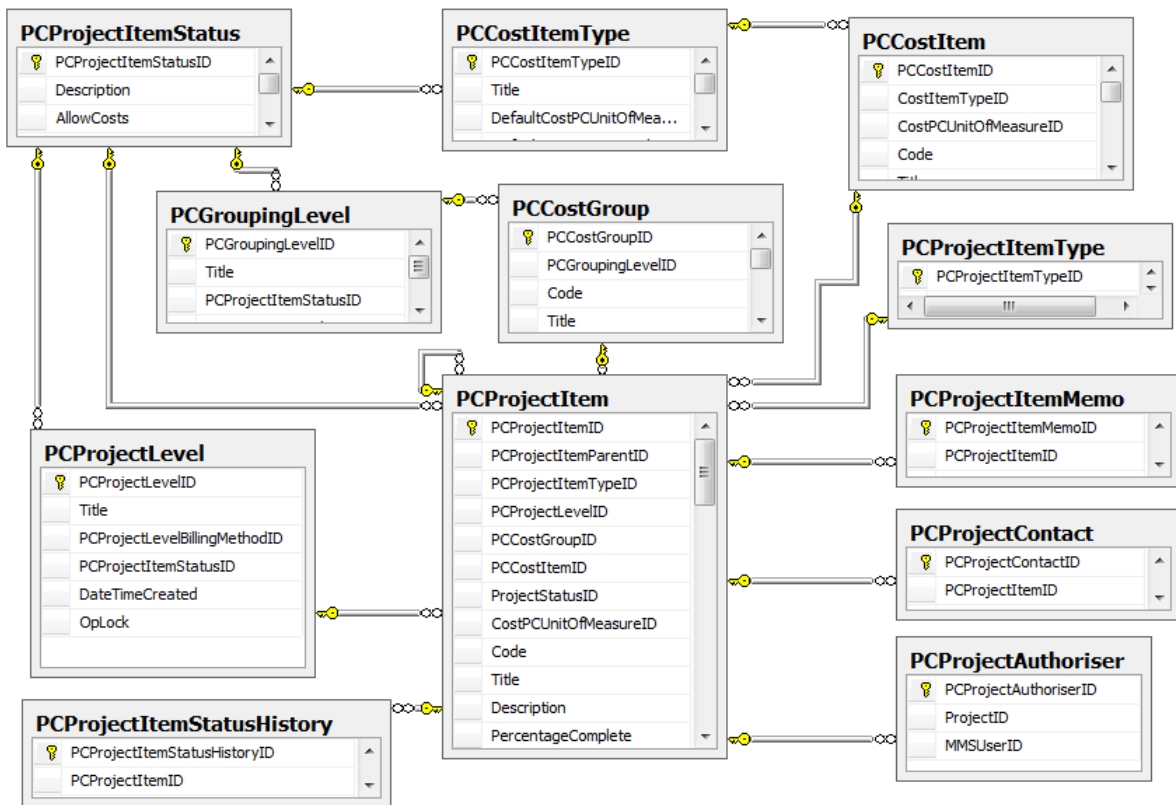
If this PCProjectItem it is a Project then the PCProjectLevelId indicates the project 'type'. If it is a Group the PCGroup indicates the group it is an instance of, and if it is an Item then the PCCostItem indicates the item it is an instance of.

The PCGroup further links to PCGroupingLevel for the 'type' of group and PCCostItem links to PCCostItemType for the 'type' of cost item. The data in these tables is created and amended in the following areas:

[Project Accounting > Utilities > System Setup > Project Structure Settings](#) and, [Project Accounting > Project Maintenance > Project Items and Groups](#).

Many of the settings that govern the behaviours of the projects, groups and items are set on these 'type' tables. Depending on the settings the PCProjectItem may have a Status, and status change history. If it is a Project it may have a Contact and an Authoriser.

Any PCProjectItem may have a linked Memo.



## Project Structure

A project is a set of PCProjectItem records that are linked in a hierarchical structure.

To enquire on the structure of a project the PCProjectItemHierarchy table should be the starting point. This has multiple links to the PCProjectItem table to allow efficient querying of the project structure.

The BelongsToPCProjectItemID will link to **many** PCProjectItems, depending on where in the structure the PCProjectItem sits.

For example:

Project24

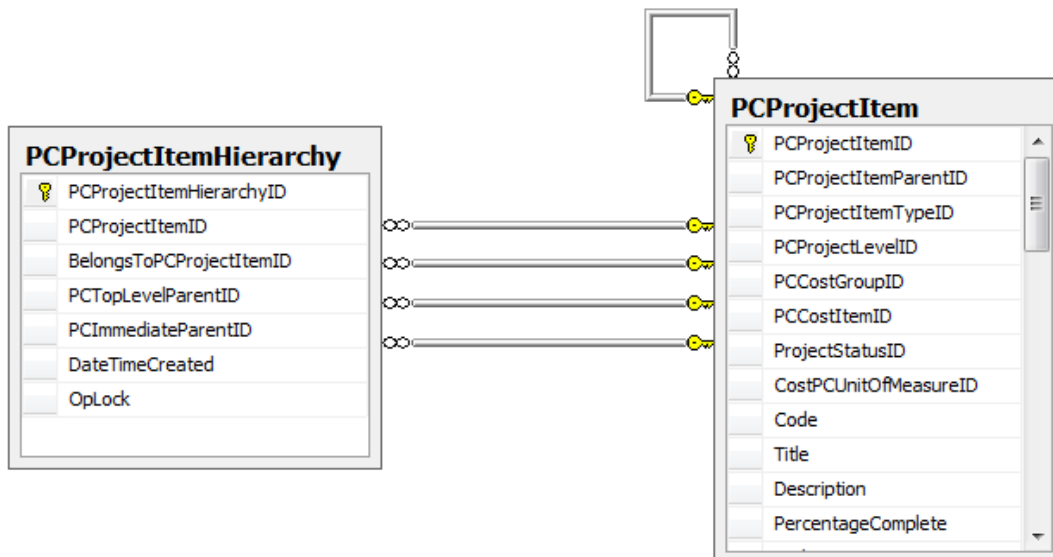
-- Group2

-- Item5

Item5 will link via the BelongsToPCProjectItemId to 3 PCProjectItems: Itself (Item5), to Group2 and to Project24.

The PCImmediateParentID links to the PCProjectItem that is the *Project* immediately above the PCProjectItem in the structure. The PCTopLevelParentID field links to the top-most project in the structure.

**Note:** These may be the same project if the PCProjectItem being queried belongs directly to the topmost project.



## Integration and Display Feature Settings

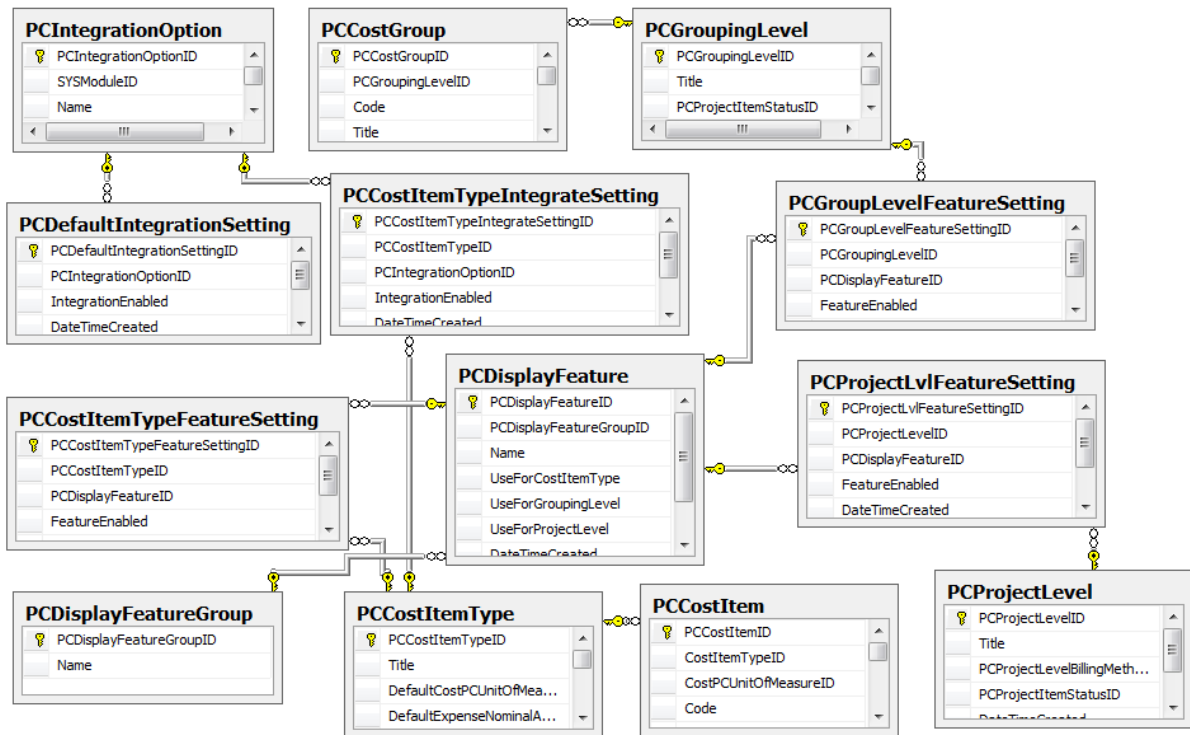
The integration settings in Project Accounting allow the user to specify how they want to operate and control the Project Accounting integration with the other modules. All possible integration options are held in PCIntegrationOption – this table has links to SYSModule to resolve which module the integration option specifically effects.

Integration is defined at two levels:

- for the overall company in PCDefaultIntegrationSetting (which is set up in the application in [Project Accounting > Utilities > System Setup > Project Accounting Settings](#))
- and per cost item type in PCCostItemTypeIntegrateSetting. This is amended in [Project Accounting > Utilities > System Setup > Project Structure Settings > Project Item Types](#)

The Display feature settings allow the User to customise Project Accounting to disable or enable specific application features at various levels: project levels, group levels and cost item types. These are accessed using [Project Accounting > Utilities > System Setup > Project Structure Settings](#) then choosing, [Project Item Types](#), [Grouping Levels](#) or [Project Levels](#).

Three link tables exist to provide this functionality: PCProjectLvlFeatureSetting, PCGroupLevelFeatureSetting and PCCostItemTypeFeatureSetting.



## Additional Information Fields

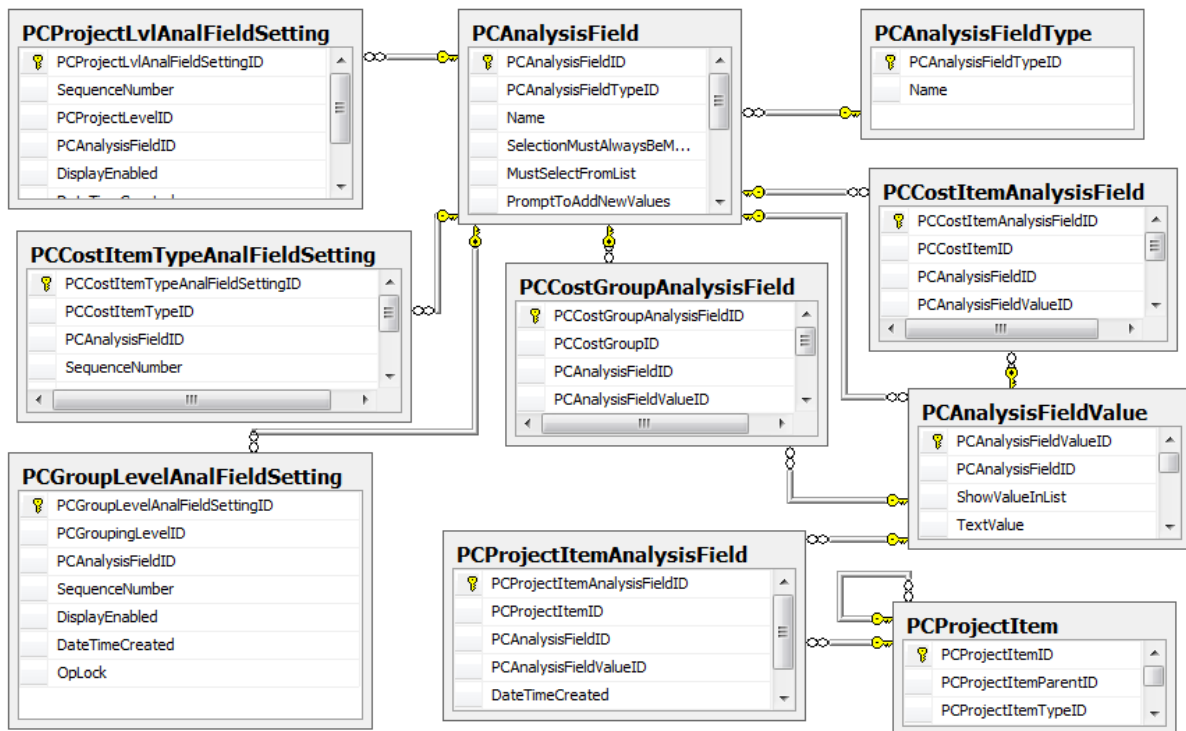
Additional Information allows project managers to add extra information to any part of a project.

The fields themselves are set up in [Project Accounting > Project Maintenance > Additional Information](#) and held in the PCAnalysisField table. If the values are only selectable from a list then the list associated with each PCAnalysisField are held in the PCAnalysisFieldValue. However if values can be entered 'free-hand' then any value entered in the application will also be held in the PCAnalysisFieldValue records.

The flexible nature of the type of information needed to be recorded means that the Additional Information values (stored in the PCAnalysisFieldValue table) can be anything from the following types: Text, Numeric Date, Boolean (yes/no), Time, Supplier, Customer, Human Resource, Stock Item. In the cases where the value is an ID to another table, the value stored in PCAnalysisFieldValue is a foreign key to the appropriate table – PLSupplierAccount for Supplier, SLCustomerAccount for Customer, TSHumanResource for Human Resource and StockItem for StockItem.

Allowable information fields can be defined for use with each: Project level, Group Level and Cost Item Type. These are maintained using [Project Accounting > Utilities > System Setup > Project Structure Settings](#) then choosing [Project Item Types](#), [Grouping Levels](#) or [Project Levels](#). These are held in the PCCostItemTypeAnalFieldSetting, PCGroupLevelAnalFieldSetting and PCProjectLvlAnalFieldSetting tables.

The additional information fields and their value are then associated to specific instances of Projects, Groups and Project items using the PCProjectItemAnalysisField table.



## Project Transactions

Postings to Project Accounting can be made from all other modules. The source of the posting is described in PCProjectEntryDescriptor, which contains a full static list of the areas a posting can originate from in the application. To obtain further information about a transaction, this table can also be used as part of a join. For example, if a project entry has been created through the raising of a Purchase Order, the following SQL could be used to resolve the purchase order associated with a particular project entry.

```
SELECT A.POPOrderReturnID
FROM POPOrderReturn AS A
INNER JOIN POPOrderReturnLine AS B ON A.POPOrderReturnID = B.POPOrderReturnID
INNER JOIN PCProjectEntry AS C ON B.POPOrderReturnLineID = C.SourceEntryID
INNER JOIN PCProjectEntryDescriptor AS D ON C.PCProjectEntryDescriptorID =
D.PCProjectEntryDescriptorID
WHERE
D.[Key] = 'Purchase Order'
```

We can see in the example above that the SourceEntryID field in PCProjectEntry is a 'weak' foreign key (in that its value is not enforced through referential integrity). The SourceEntryID value can be a foreign key to a variety of tables that can be correctly resolved by linking to the PCProjectEntryDescriptor table and filtering on the required area.

There are two overall types of entry that can exist in PCProjectEntry – these are 'Actual' and 'Potential' – this information can be obtained by linking PCProjectEntry to the PCEntryType enumeration table. The example below describes the differences between actual and potential entries and when they would exist.

When a purchase order is raised for each line in the order, a potential project entry is created. If the order is to be despatched immediately or when stock is allocated an actual entry will be created which will have the potential entry as its parent (through the PCProjectEntry's ParentEntryID – PCProjectEntryID self-referring link). The potential entry will have its total reduced to match the actual entries total. Once the order has been completely fulfilled, the potential entries total will be reduced to zero but will not be deleted.

The PCProjectEntryPosting table is used in the area of Billing. When a Bill is created, PCProjectEntryPosting entries are made against specific project costs (which are held as PCProjectEntry's). As a cost is resolved a series of PCProjectEntryPosting's can be made until cost is completely paid. This is the reason for the **one-to-many** relationship between PCProjectEntry and PCProjectEntryPosting.



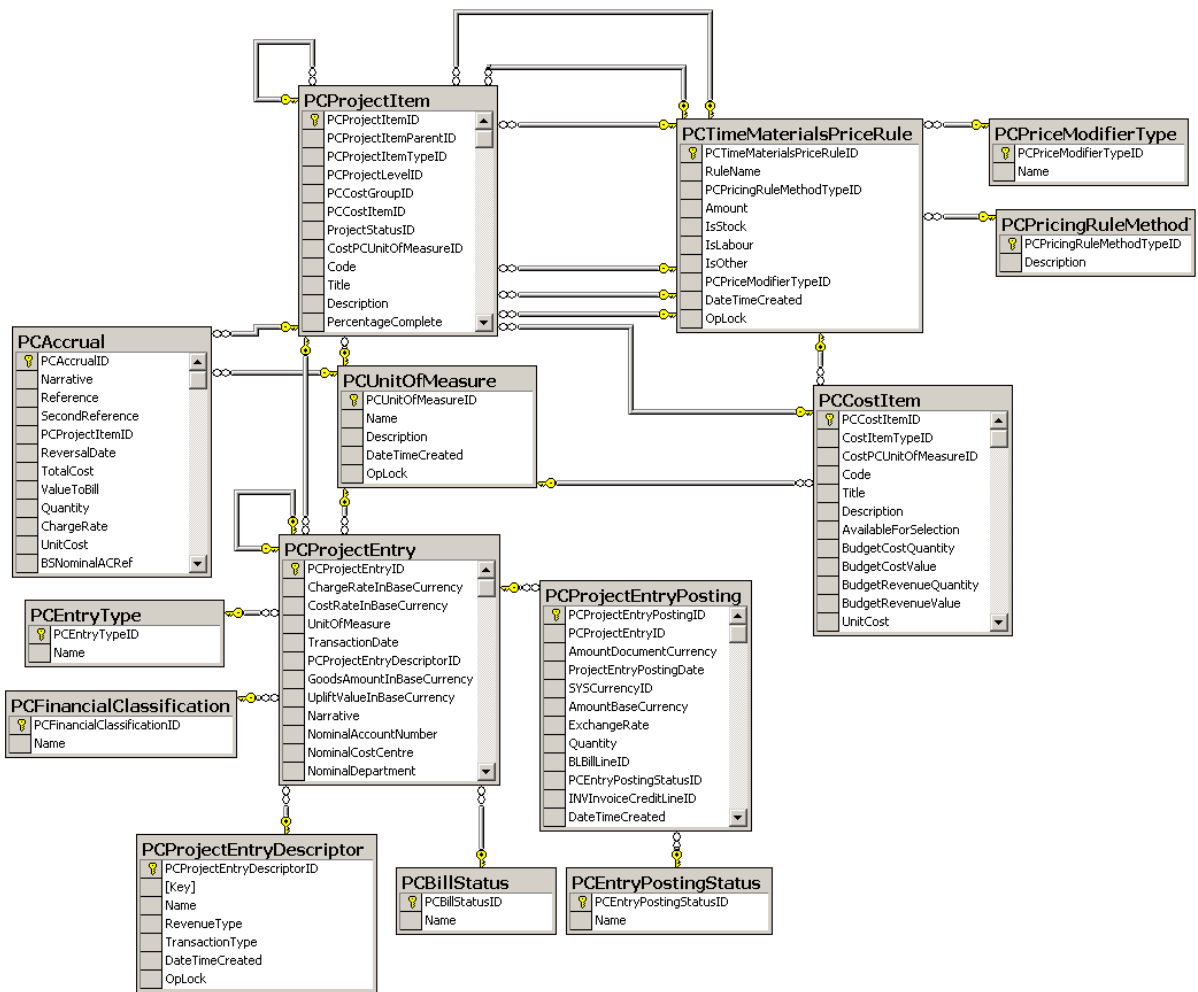
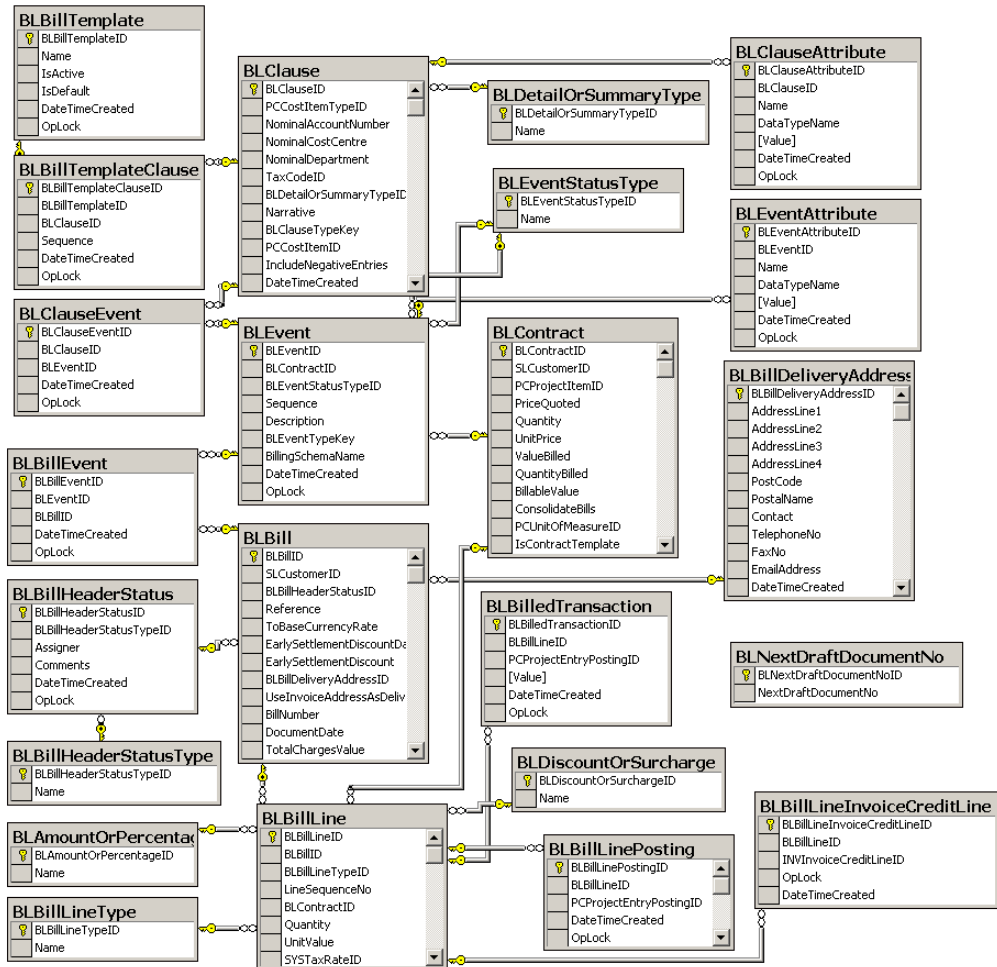


Diagram: Project Transactions

## Billing

Project Accounting allows the user to produce bills for project customers. Bills are an indication of how much a customer will owe you, typically for costs (direct and/or indirect) relating to a project. When a bill is created, the header details are stored in the BLBill table, this has a link to BLContract (which then links to SLCustomerAccount) to indicate the customer this bill is to be created for.

Once a bill has been created, bill lines can then be added to it. These are held in the BLBillLine table and can be one of three types (specified in the BLBillLineType enumeration table) – Standard Bill Line, Comment Line – Private and Comment Line – Public. A standard bill line represents a value that is to be billed (or refunded) against a specified project. Both types of comment line provide a means of adding information to a bill. If it is a private comment, it is not printed on documentation.



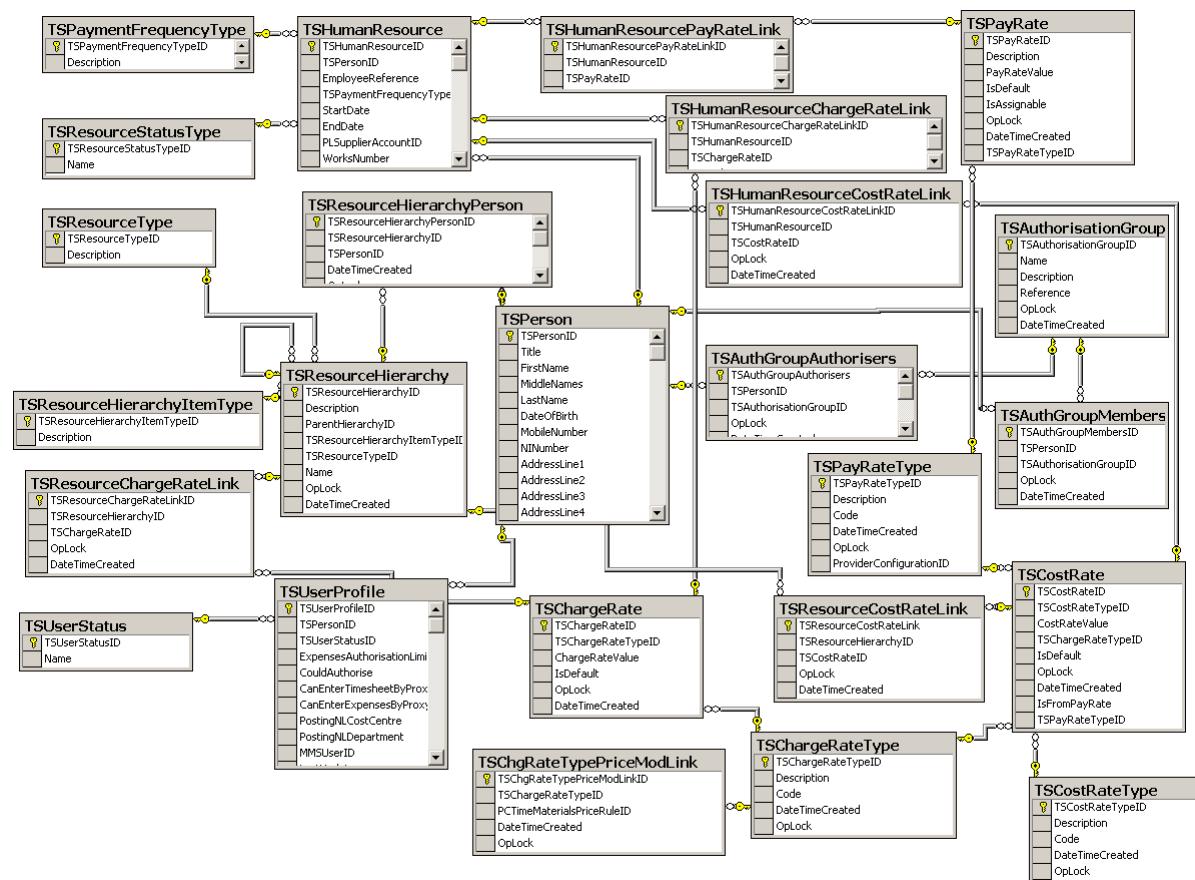
# Timesheets and Expenses

## People and Resources

To enter timesheets and expense claims, timesheet users must first be set up (this can be done in the application at **Project Accounting > Project Maintenance > Resources**). The details of each individual (or resource) are held in the TSPerson and TSHumanResource tables.

To support the concept of assigning people to authorise expense claims there are links from TSPerson to TAuthGroupAuthorisers, TAuthorisationGroup and TAuthGroupMembers. This relationship allows many authorisers to authorise many people or resources.

Once a resource has been created, that resource can be added to a hierarchy (Project Accounting > Project Maintenance > Hierarchy) to support the assignment of cost and pay rates on a group by group basis. The hierarchy is stored in TResourceHierarchy which contains a self-referring link to maintain the hierarchy tree structure.

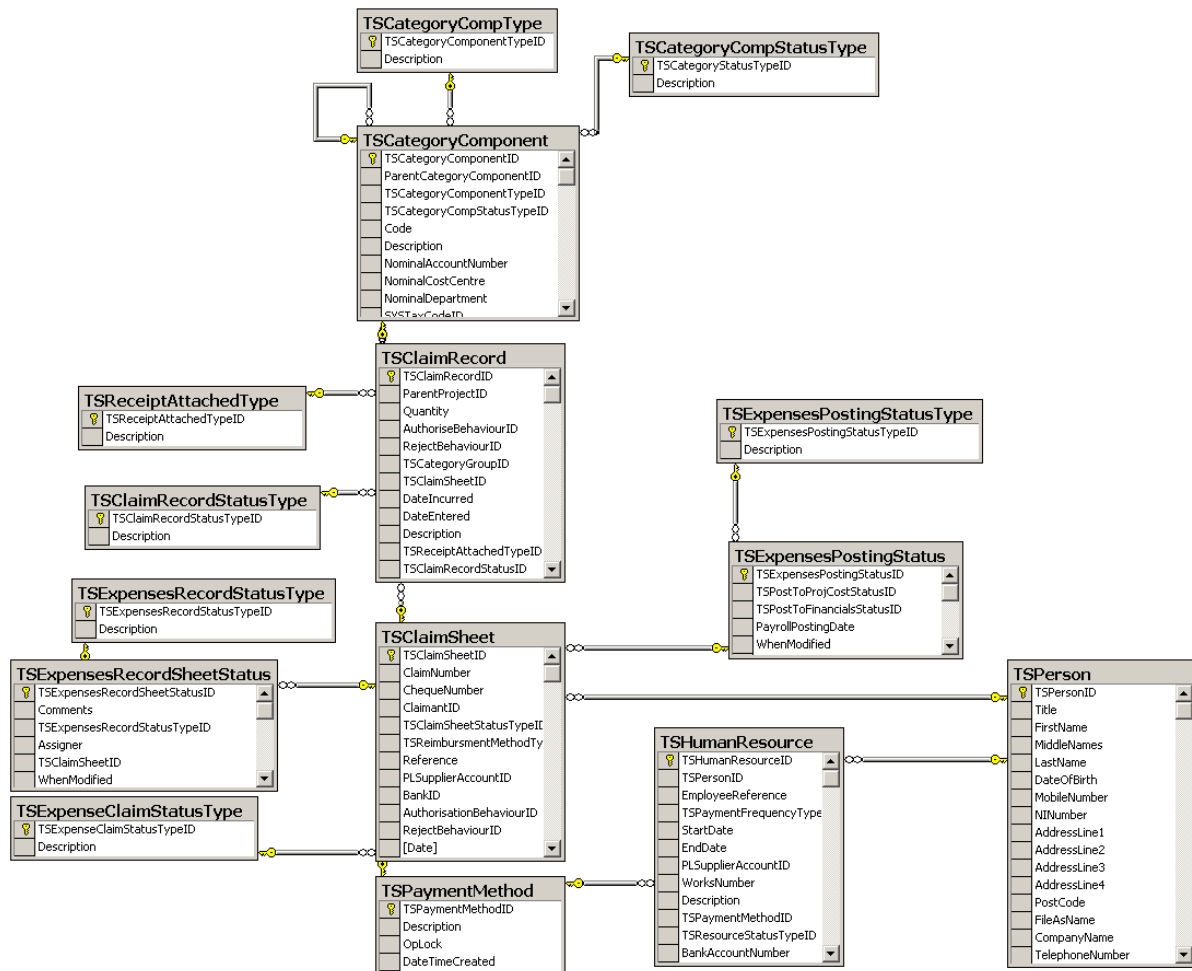


## Expense Claims

Expense claims are a means of entering and processing 'personal' expenses as required. Expense claims are recorded by timesheet users (resources), these claims are then charged to activities – which may be project items from project accounting or locally created activities. They can be entered through [Project Accounting > Enter Transactions > Enter Expense Claims](#).

The 'header' information for the claim, such as the current timesheet user is held in the TSClaimSheet table. Expense 'lines' for the associated header are then held in the TSClaimRecord table, there is a one-to-many relationship between the TSClaimSheet and TSClaimRecord tables.

The TSCategoryComponent table is used to maintain the list of expense category groups and items and has a self-referential link for the association between groups and items. To differentiate between items and groups there is a link to the TSCategoryCompType enumeration table.



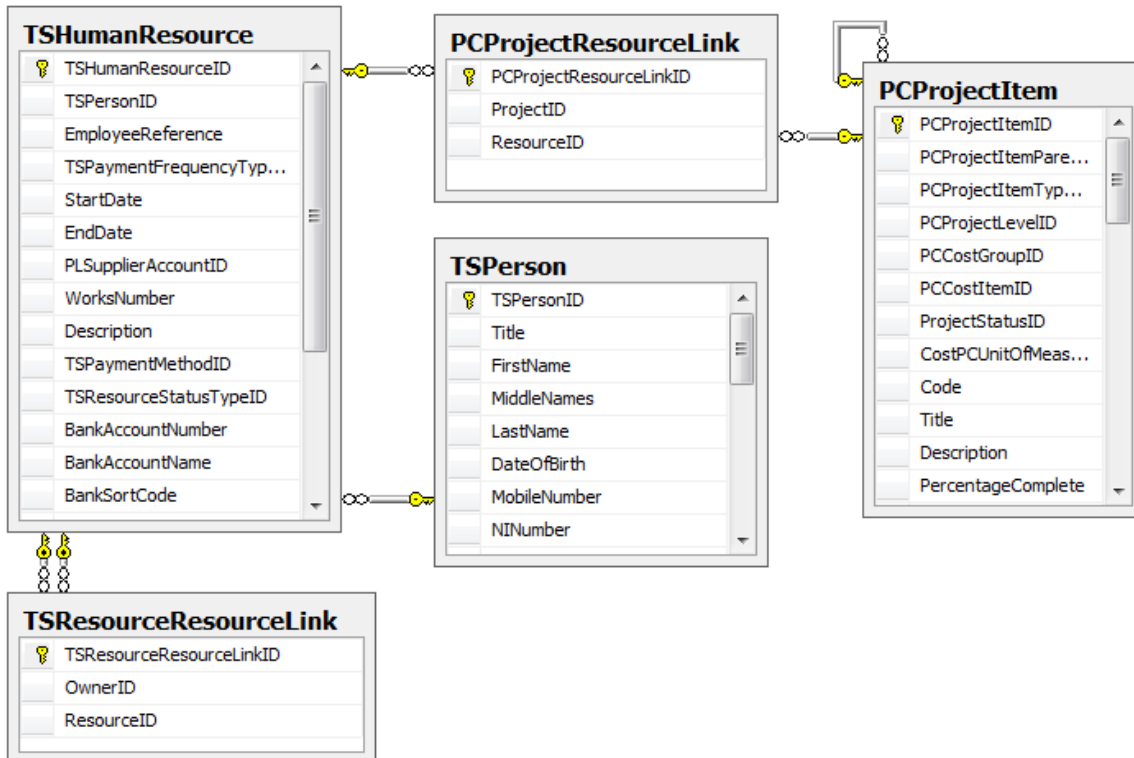


## Project and Resource filtering

In version 2010 the Project and Resource filtering features were introduced.

- Project filtering allows resources to be linked to just certain projects so that when they enter timesheets or expenses only those projects are available for selection.
- Resource filtering enables links to be created between resources such that one resource may be able to enter time and expenses on behalf of other resources or only for themselves.

The PcProjectResourceLink and TSResourceResourceLink records provide this functionality.



## Look-up Tables

These are tables that are set up by default in all datasets, whether new or converted. These cannot be amended by the user. They contain sets of values that the system uses to control its functionality, such as statuses and types.

For example; OrderReturnLineType is selected by the user when entering a new order line. The screen they are presented with and the data they must complete depends upon which type of line they select.

Many data tables will contain the relevant 'ID' of records in one or more of these look-up tables but to find out what that ID means you would have to link to the relevant table itself and read the 'name' field from the associated look-up record.

To avoid this, the look-up tables and their values are listed here as a quick way to access this information. They are listed in alphabetic order by table name.

### AuthorisationStatus

AuthorisationStatusId	Name
0	Not required
1	Not Authorised
2	Authorised
3	Queried

### BackToBackStatus

BackToBackStatusId	Name
0	PO Not Required
1	PO Required
2	PO Not Generated
3	PO Generated
4	Fulfilled

### BLAmountOrPercentage

BLAmountOrPercentageID	Name
0	Amount
1	Percentage

### BLBillHeaderStatusType

BLBillHeaderStatusTypeID	Name
0	Draft
1	Not Saved
2	Printed
3	Temporary
4	Finalised

### BLBillLineType

BLBillLineTypeID	Name
0	Value Line
1	Private Comment Line
2	Public Comment Line

### BLDetailOrSummaryType

BLDetailOrSummaryTypeID	Name
0	Detail
1	Summary

### BLDiscountOrSurcharge

BLDiscountOrSurchargeID	Name
0	Discount
1	Surcharge



### BLEventStatusType

<b>BLEventStatusTypeID</b>	<b>Name</b>
0	Held
1	Complete
2	Force
3	Evaluate

### BomBuildPackageType

<b>BomBuildPackageTypeID</b>	<b>Name</b>
0	Build
1	Salvage

### BomBuildSessionStatus

<b>BomBuildSessionStatusId</b>	<b>Name</b>
0	Built
1	Built Cannot Reverse

### BomBuildSessionType

<b>BomSessionTypeID</b>	<b>Name</b>
0	Unspecified
1	Record Build
2	Confirm Build

### BomComponentLineType

BomComponentLineTypeId	Name
0	Component
1	Phantom
2	Subassembly
3	Comment
4	Document

### BomCostType

BomCostTypeId	Name
0	Component
1	Bill of Material
2	Operation
3	Subcontract

### BomOperationLineType

BomComponentLineTypeId	Name
0	Operation
1	Subcontract
2	Comment
3	Document

**BomVersionStatus**

<b>BomVersionStatus</b>	<b>Name</b>
0	New
1	Active
2	Obsolete
3	Retired
4	Hidden

**CBAccountLocationType**

<b>CBAccountLocationTypeID</b>	<b>Name</b>
0	Main

**CBAccountType**

<b>CBAccountTypeID</b>	<b>Name</b>
0	Current
1	Deposit

### CBDirectDebitDocumentType

<b>CBDirectDebitDocumentTypeID</b>	<b>Name</b>
0	SalesReceiptStandingOrder
1	PurchasePaymentStandingOrder
2	NominalPaymentWithoutTaxStandingOrder
3	NominalReceiptWithoutTaxStandingOrder
4	NominalPaymentWithTaxStandingOrder
5	NominalReceiptWithTaxStandingOrder
6	SalesReceiptDirectDebit
7	PurchasePaymentDirectDebit
8	NominalPaymentWithoutTaxDirectDebit
9	NominalReceiptWithoutTaxDirectDebit
10	NominalPaymentWithTaxDirectDebit
11	NominalReceiptWithTaxDirectDebit

### CBDirectDebitTranType

<b>CBDirectDebitTranTypeID</b>	<b>Name</b>
0	BP DD
1	BP SO
2	BR DD
3	BR SO

### CBPaymentFrequency

CBPaymentFrequencyId	Name
0	Month
1	Quarterly
2	Other

### CBTranType

CBTranTypeID	Name
0	Unknown
1	Opening Balance
2	Receipt
3	Payment
4	Direct Debit
5	Standing order
6	Cancelled Payment
7	Cancelled Receipt

### ConfirmationIntentType

ConfirmationIntentTypeId	Name
0	Confirm via Despatch/Receipt
1	Confirm
2	No Confirmation

### CostingMethod

CostingMethodId	Name
0	FIFO
1	Average
2	Standard
3	Actual

### CustomerType

CustomerTypeId	Name
0	Cash
1	Credit
2	Prospect

### DefaultPricingSourceType

DefaultPricingSourceTypeId	Name
0	Last Buying Price
1	List Price

### DiscountType

DiscountTypeId	Name
0	Percentage
1	Amount
2	Fixed price

## DocumentPrintStatus

DocumentPrintStatusId	Name
0	Not Printed
1	Printed
2	Amended Since Last Printed

## DocumentStatus

DocumentStatusId	Name
0	Live
1	On Hold
2	Completed
3	Dispute
4	Cancelled

## EntrySource

EntrySourceId	Name
0	Stock
1	SOP
2	POP
3	BOM
4	Manual

## EventLogType

EventLogTypeId	Name
0	Information
1	Exception
2	Error

## IntrastatDateType

IntrastatDateTypeId	Name
0	Despatch/Receipt Date
1	Invoice Date

## IntrastatDestinationType

IntrastatDestinationTypeId	Name
0	Despatch
1	Receipt
2	Both

## IntrastatEntryHeaderStatus

IntrastatEntryHeaderStatusId	Name
0	Legacy
1	Unsubmitted
2	Produced



### IntrastatStatus

IntrastatStatusId	Name
0	Not Required
1	Required
2	Completed
3	In Progress
4	Deleted

### INVCustomerUnitPricePref

INVCustomerUnitPricePrefId	Name
0	Tax Exclusive
1	Tax Inclusive

### INVInvoiceCreditType

INVInvoiceCreditTypeId	Name
0	Invoice
1	Credit

### INVInvoiceStatus

INVInvoiceStatus	Name
0	Draft
1	Posted
2	Ready
3	Printed
4	Cancelled

## LandedCostsType

LandedCostsTypeId	Name
0	Not Applicable
1	Percentage based
2	Value Based

## ModeOfTransport

ModeOfTransportId	Code	Description	CountryCode
0	1	Sea	IE
1	2	Rail	IE
2	3	Road	IE
3	4	Air	IE
4	5	Post	IE
5	7	Fixed Transport Installation	IE
6	8	Inland Waterway	IE
7	9	Own Propulsion	IE

## MovementBalanceType

MovementBalanceTypeId	MovementBalanceTypeName
0	Confirmed
1	Not Confirmed
2	Shortfall
3	Unconfirmed Return

## NatureOfTransactionCode

NatureOfTransactionCodeId	Code	CountryCode
0	10	GB
1	16	GB
2	17	GB
3	18	GB
4	20	GB
5	30	GB
6	40	GB
7	50	GB
8	60	GB
9	70	GB
10	80	GB
11	90	GB
12	1	IE
13	2	IE
14	3	IE
15	4	IE
16	5	IE
17	6	IE
18	7	IE
19	8	IE
20	9	IE

### NLAccountReportType

NLAccountReportTypeId	Name
0	Balance Sheet
1	Profit and Loss

### NLAccountType

NLAccountTypeId	Code	Name
0	P	Posting
1	G	Group
2	M	Memorandum

### NLAnnualBudgetApportProfile

NLAnnualBudgetApportProfileId	Name
0	Budget Profile 1
1	Budget Profile 2
2	Budget Profile 3

### NLAnnualBudgetTitle

NLAnnualBudgetTitleId	YearRelativeToCurrent	BudgetTitle
0	1	Next Year
1	0	Current Year
2	-1	Last Year
3	-2	2 Years Ago
4	-3	3 Years Ago
5	-4	4 Years Ago
6	-5	5 Years Ago

### NLAnnualBudgetType

NLAnnualBudgetTypeid	Name
0	No budget
1	Monthly Budget
2	Annual Budget
3	Group Budget
4	Profiled Budget

### NLFinancialReportLineType

NLFinancialReportLineTypeid	Name
0	Text
1	Sub Total
2	Account Group (Single)
3	Account Group (Multiple)
4	Current Period Profit

### NLJournalTemplateType

NLJournalTemplateTypeid	Name
0	Accounts only
1	Percentage
2	Value

### NLJournalTranTaxType

NLJournalTranTaxTypeId	Name
0	Not Applicable
1	Input Goods
2	Input Tax
3	Output Goods
4	Output Tax

### NLJournalType

NLJournalTypeId	Name
0	Recurring Entry
1	Reversing Entry
2	Single Entry

### NLLayoutDebitOrCreditType

NLLayoutDebitOrCreditTypeId	Name
0	Unknown
1	Debit
2	Credit

### NL LayoutPositionType

NLLayoutPositionTypeId	Name
0	Unknown
1	Left
2	Right

### NLNominalTranType

NLNominalTranTypeId	Name
0	General
1	Manual
2	Prepayment
3	Accrual
4	Previous Year
5	Tax Payment
6	Tax Receipt
7	Payment
8	Receipt

### NLPendNominalAccountType

NLLayoutPositionTypeId	Name
0	Budgets Only
1	New Account

### NominalUsage

NominalUsageld	NominalusageName
0	Stock
1	Revenue
2	Accrued Receipts
3	Issues

## OrderReturnLineType

OrderReturnLineTypeId	Name
0	Standard Item
1	Free Text Item
2	Additional Charge
3	Comment Line

## PaymentType

PaymentTypeId	Name
0	Full
1	Part

## PCAnalysisFieldType

PCAnalysisFieldTypeId	Name
0	Text
1	Numeric
2	Date
3	Time
4	Yes/No
5	Supplier List
6	Customer List
7	Product List
8	Timesheet User List



### PCBillStatus

PCBillStatusId	Name
0	Draft
1	Finalised
2	Not Billable
3	Not Billed

### PCBudgetLevel

PCBudgetLevelId	Name
0	Project only budgets
1	Project and group
2	Project, group and cost item budgets

### PCConfigurableTerm

PCConfigurableTermID	ConfiguredTerm	DefaultTerm
0	Project	Project
1	Group	Group
2	Project Item	Project Item
3	Project Level	Project Level
4	Grouping Level	Grouping Level
5	Project Item Type	Project Item Type
6	Customer	Custoemr
7	Quantity	Quantity

### PCCreateProjectPreference

PCCreateProjectPreferenceId	Name
1	Create a blank project
2	Copy a project template
3	Copy an existing project
4	Create from a Quotation
5	Create from a Sales Order
6	Copy and Archived Project

### PCDisplayFeatureGroup

PCDisplayFeatureGroupId	Name
0	Record general details
1	Display costing information
2	Display charging information

### PCEntryPostingStatus

PCEntryPostingStatusId	Name
0	Available for refund
1	Created

### PCEntryType

PCEntryTypeId	Name
0	Actual
1	Potential

### PCFinancialClassification

PCFinancialClassificationId	Name
0	Cost
1	Revenue

### PCOverheadUpliftModifier

PCOverheadUpliftModifierId	Name
0	Percentage
1	Amount
2	Not Applicable

### PCPriceModifierType

PCPriceModifierId	Name
0	Default
1	User Defined
2	Read Only

### PCPricingRuleMethodType

PCPricingRuleMethodTypeId	Name
0	Not Applicable
1	Cost + %
2	Cost + Value
3	Set Value
4	Timesheets charge rate
5	Standard Selling Price
6	Cost

## PCProjectArchiveStatus

PCProjectArchiveStatus	Name
0	Live
1	Archived

## PCProjectEntryDescriptor

PCProjectEntryDescriptorId	Key	Name	RevenueType	TransactionType
0	Sales Invoice	Sales Invoice	True	SLI
1	Sales Credit	Sales Credit	True	SLC
2	Purchase Invoice	Purchase Invoice	False	PLI
3	Purchase Credit	Purchase Credit	False	PLC
4	Nominal Payment	Nominal Payment	False	NLP
5	Nominal Receipt	Nominal Receipt	False	NLR
6	Stock Return	Stock Return	False	STR
7	Stock Issue	Stock Issue	False	STI
8	Cost Adjustment	Cost Adjustment	False	PCA
9	Cost Accrual	Cost Accrual	False	PAC
10	Timesheet Entry	Timesheet Entry	False	TSE
11	Expense Entry	Expense Entry	False	EXP
12	Revenue Adjustment	Revenue Adjustment	True	PRA
13	Stock Allocation	Stock Allocation	False	STA
14	Purchase Order	Purchase Order	False	POO
15	Purchase Return	Purchase	False	POR

		Return		
16	Sales Order	Sales Order	True	SOO
17	Sales Quote	Sales Quote	True	SOQ
18	PO Non Stock Received	PO Non Stock Received	False	PSR
19	PO Non Stock Returned	PO Non Stock Returned	False	PSD
20	Cost Opening Balance	Cost Opening Balance	False	COB
21	Revenue Opening Balance	Revenue Opening Balance	True	ROB
22	Sales Batch Invoice	Sales Batch Invoice	True	SBI
23	Sales Batch Credit Note	Sales Batch Credit Note	True	SBC
24	Purchase Batch Invoice	Purchase Batch Invoice	True	PBI
25	Purchase Batch Credit Note	Purchase Batch Credit Note	True	PBC

### PCProjectItemStatus

PCProjectItemStatusId	Description	AllowCosts	AllowRevenue	AllowDeletion
1	<Not Applicable>	1	1	1
2	Bid	1	1	0
3	Complete	0	0	0
4	Delete	0	0	1
5	In Progress	1	1	0
6	Inactive	1	0	0
7	On Hold	1	1	0
8	Open	1	1	0

### PCProjectItemType

PCProjectItemTypeId	Name
0	Project
1	Group
2	Item
3	Project Template

### PCProjectLevelBillingMethod

PCProjectLevelBillingMethodId	Name
0	Time & Materials
1	Fixed Price
2	Costs Plus
3	Non-Productive

## PLAllocationType

PLAllocationTypeId	Name
0	Manual Payment
1	Automatic Payment
2	Reverse Posting
3	Small Values Write Off
4	Manual Remittance Cheque
5	Contra Entry
6	Automatic Remittance Cheque
7	Manual Notification
8	Manual Opening Balance
9	Manual Receipt

## PLPaymentDocumentTypeId

PLPaymentDocumentTypeId	Name
0	Remittance and Cheque
1	Remittance Only (Paper)
2	Cheque Only
3	Electronic Payment
4	Electronic Remittance and Payment (Paper)
5	Remittance only (Email)
6	Electronic Payment and Remittance (Email)

### POPAuthAuditEventType

POPAuthAuditEventTypeId	Name
0	Rule
1	Order Created
2	Order Amended
3	Order Authorised
4	Order Printed
5	Query Respond
6	Notify
7	Super Authority
8	Alternative Authority
9	Authoriser Amended

### POPAuthPrincipalContentType

POPAuthPrincipalContentTypeId	Name
0	Anyone
1	User
2	Role
3	Other
4	Unknown

### POPAuthRoleType

POPAuthRoleTypeId	Name
0	Authoriser
1	Originator



### POPIvoiceCreditType

POPIvoiceCreditTypeld	Name
0	Invoice
1	Credit
2	T e-mail Invoice
3	T e-mail Credit

### POPOrderGenerationType

POPOrderGenerationTypeld	Name
0	Use Generate Orders
1	Use MRP
2	Use Both Methods

### POPOrderReturnLineAddrType

POPOrderReturnLineAddrTypeld	Name
0	None
1	Customer
2	Supplier
3	Free Text
4	Other

### POPOrderReturnType

POPOrderReturnTypeld	Name
0	Purchase Order
1	Purchase Return

### POPRcptRtnLineSource

POPRcptRntLineSourceId	Name
0	GRN
1	Invoice or Credit

### POPReceiptReturnType

POPReceiptReturnTypeId	Name
0	Receipt
1	Despatch

### POPReorderLineStatus

POPReorderLineStatusId	Name
0	Purchase
1	Hold
2	Reject

### POPToReorderItemStatus

POPToReOrderItemStatusId	Name
0	None
1	Part
2	All

### PSCondOperatorType

PSCondOperatorTypeid	Name
0	Equal
1	Greater Than
2	Greater Than Or Equal
3	Less Than
4	Less Than Or Equal
5	Like
6	Not Equal
7	Not Like

### PSLogicalOperatorType

PSLogicalOperatorTypeid	Name
0	And
1	Or

### ShowOnPickingListType

ShowOnPickingListType	Name
0	Do Not Show
1	Show As usual
2	Show Though Excluded

### SLAllocationType

SLAllocationTypeId	Name
0	Manual Receipt
1	Write Off
2	Reverse Posting
3	Automatic Allocation
4	Small Values Write Off
5	Contra Entry
6	Free text Invoice
7	Reverse Finance Charge
9	Manual Notification
9	Manual Opening Balance
10	Manual Payment

### SLDateToUseForAgeing

SLDateToUseForAgeingId	Name
0	Use Instrument Date
1	Use Due Date
2	Use Ledger Period

### SLDocumentMessageUsedFor

LDocumentMessageUsedFor	Name
0	Statements and Letters
1	Letters
2	Statements

### SLOfficeType

SLOfficeTypeld	Name
0	Independent
1	Branch
2	Head Office
3	Independent No Statement
4	Head Office No Statement

### SLReportMessageType

SLReportMessageTypeld	Name
0	Both
1	Statements
2	Letters

### SLTrantoShowOnStatements

SLTranToShowOnStatementsId	Name
0	All transactions
1	Outstanding transactions only
2	Outstanding plus all from a specified date

### SOPDespatchReceiptType

SOPDespatchReceiptTypeld	Name
0	Despatch Note
1	Receipt

### SOPInvoiceCreditType

SOPInvoiceCreditTypeId	Name
0	Invoice
1	Credit Note
2	T e-mail Invoice
3	T e-mail Credit
4	Tax Only Invoice
5	Tax Only Credit

### SOPOrderEntryType

SOPOrderEntryTypeId	Name
0	Full Order Entry
1	Rapid Order Entry
2	Trade Order Entry

### SOPOrderFulfilmentMethod

SOPOrderFulfilmentMethod	Name
0	From Stock
1	Supplier via Stock
2	Direct to Customer

### SOPOrderReturnType

SOPOrderReturnTypeld	SOPOrderReturnTypeName
0	Sales Order
1	Sales Return
2	Pro Forma Invoice
3	Quotation
4	Repeat Template
5	Tax Only Order
6	Tax Only Return
7	T e-mial Order
8	T e-mail Return

### SOPPartialAllocationType

SOPPartialAllocationTypeld	Name
0	Always
1	Full Order Line
2	Full Order

### SOPRepeatOrderType

SOPRepeatOrderTypeld	Name
0	Specified Count
1	Specified Expiry Date
2	Indefinite

## SourceAreaType

SourceAreaType	SourceAreaTypeName
0	Supplier
1	Customer
2	Internal
3	Write Off
4	Opening Balance
5	BOM

## STKAutoGenerationOptionType

STKAutoGenerationOptionTypeId	Name
0	No automatic number generation
1	Use sequence numbers
2	Use data and sequence numbers
3	User order number and sequence number

## STKAutoGenerateSeparator

STKAutoGenerateSeparatorId	Name	SeparatorCharacter
1	None	
2	Space	
3	Colon	:
4	Forward Slash	/
5	Backward Slash	\
6	Dot	.
7	Hyphen	-



### STKFulfilmentSequenceType

STKFulfilmentSequenceTypeId	Name
0	Bin Priority
1	Smallest Quantity First
2	Largest Quantity First
3	Oldest Sell By Date
4	Oldest Use By Date
5	Earliest FIFO Movement Date

### STKLabelPrintingOptionType

STKLabelPrintingOptionTypeId	Name
0	Label not required
1	Label required per item
2	Label required per batch

### STKShelfLifeType

STKShelfLifeTypeId	Name
0	Days
1	Weeks
2	Months
3	Years

### StockItemStatus

StockItemStatusId	StockItemStatusName
0	Active
1	Inactive

## StockItemType

StockItemTypeId	StockitemTypeName
0	Stock
1	Miscellaneous
2	Service/Labour
3	Undefined
4	SOPItemDespatched
5	SOPItemNotDespatched

## StockTake Selector

StockTakeSelectorId	Name
0	Average buying price
1	Total value of stock

## StockTake Status

StockTakeStatusId	Name
0	No stocktake
1	Selecting items
2	Sheets printed
3	Entering counts
4	Complete

## StockTakeType

StockTakeTypeId	Name
0	By Stock Item
1	By Bin

### SYSAccountMemoType

<b>SYSAccountMemoTypeid</b>	<b>Name</b>
0	Default
1	CreditInfo

### SYSAccountType

<b>SYSAccountTypeid</b>	<b>Name</b>
0	Balance Forward
1	Open Item
2	Auto Allocation

### SYSAddressContactUse

<b>SYSAddressContactUseid</b>	<b>Name</b>
0	Company's main address

## SYSAuditTrailEntryType

SYSAuditTrailEntryTypeId	StockitemTypeName
0	Unknown
1	Sales Receipt
2	Sales Payment
3	Sales Invoice
4	Sales Credit Note
5	Sales Opening Balance Invoice
6	Sales Opening Balance Credit Note
7	Purchase Receipt
8	Purchase Payment
9	Purchase Invoice
10	Purchase Credit Note
11	Purchase Opening Balance Invoice
12	Purchase Opening Balance Credit Note
13	Nominal Journal
14	Nominal Manual Batch Journal
15	Nominal Prepayment
16	Nominal Accrual
17	Nominal Previous Year Journal
18	Nominal Bank Tax Payment Journal
19	Nominal Bank Tax Receipt Journal
20	Nominal Bank Payment Journal

21	Nominal Bank Receipt Journal
22	Vat Transaction Entry

### SYSCompanyLocationType

SYSCompanyLocationTypeld	Name
0	Main

### SYSCondOperatorType

SYSCondOperatorTypeld	Name
0	Equal
1	GreaterThan
2	GreaterThanOrEqual
3	LessThan
4	LessThanOrEqual
5	Like
6	NotEqual
7	NotLike

### SYSContactType

SYSContactTypeld	Name
0	Telephone Number
1	Fax Number
2	Email Address
3	Web Address
4	Recipient name
5	Mobile Number

## SYSDocTransmissionMethod

SYSDocTransmissionMethodId	Name
0	Post
1	Email
2	Transaction Email

## SYSDocumentType

SYSDocumentTypeId	UsedInSYSModuleID	Name	SYSTraderContactRoleID	Is Mandatory
0	1	Sales Order	1	1
1	1	Sales Invoice	1	1
2	2	Purchase Order	1	1
3	2	Purchase Invoice	1	1
4	1	Customer Statement	2	1
5	1	Sales Credit Notes	1	1
6	1	Customer Debtors Letters	1	1
7	7	Sales Despatch Notes	7	1
8	7	Sales Quotations	7	1
9	2	Purchase Credit Notes	2	1
10	2	Purchase Remittance	2	1
11	2	Purchase Return	2	1
12	7	Sales Proforma	7	1

13	5	Draft Bill	5	1
14	5	Billing Invoice	5	1
15	5	Billing Credit Note	5	1

### SYSExchangeRateAction

SYSExchangeRateActionId	Name
0	Amendability changed
1	Rate type changed
2	Single rate to Base changed
3	Single rate to Euro changed
4	Period rate to Base changed
5	Period rate to Euro changed
6	Period rate to Base deleted
7	Period rate to Euro deleted
8	New Base period rate added
9	New Euro period rate added
10	New currency added

### SYSExchangeRateAmendType

SYSExchangeRateAmendTypeId	Name
0	Not amendable
1	Amendable – cash transactions
2	Amendable – all transactions
3	Fixed to Euro

### **SYSExchangeRateType**

<b>SYSExchangeRateTypeId</b>	<b>Name</b>
0	Single
1	Period
2	Single and Period

### **SYSImportType**

<b>SYSImportTypeId</b>	<b>Name</b>
0	New
1	Deleted

### **SYSLogicalOperatorType**

<b>SYSLogicalOperatorType</b>	<b>Name</b>
0	And
1	Or

### **SYSModule**

<b>SYSModuleId</b>	<b>Name</b>
0	Not specified
1	Sales Ledger
2	Purchase Ledger
3	Nominal Ledger
4	Payroll
5	Project Accounting
6	Stock Control
7	Sales Order Processing
8	Purchase Order Processing
9	Invoicing



10	Cash Book
11	Bill of Materials
12	Fixed Assets
13	Unknown
14	Manufacturing
97	Price Book
98	Tax

### SYSModulePostingStatus

SYSModulePostingStatusId	Name
0	Open
1	Closed
2	Future

### SYSNotificationPriorityType

SYSNotificationPriorityTypeID	Name
0	Low
1	Normal
2	High

### SYSNotificationStatusType

SYSNotificationStatusTypeID	Name
0	Not started
1	In Progress
2	Completed
3	Waiting on someone else
4	Deferred

### SYSPaymentTermsBasis

<b>SYSPaymentTermsBasisID</b>	<b>Name</b>
0	Calendar monthly
1	From start of month
2	From end of month
3	From document date

### SysPeriodAction

<b>SYSPeriodActionId</b>	<b>SYSPeriodActionTypeId</b>	<b>Name</b>
0	0	Adding new accounting period
1	0	Accounting period end date changed
2	0	Accounting period deleted
3	1	Module period opened
4	1	Module period closed
5	1	Module period re-opened
6	1	Module period set to Future

### SYSPeriodActionType

<b>SYSPeriodActionTypeId</b>	<b>Name</b>
0	Accounting period
1	Module period

### SYSTaxECTerm

SYSTaxECTermId	Name
0	Not applicable
1	Sales goods and related services
2	Sales non related services
3	Purchases goods and related services
4	Purchases non related services
5	Reverse charge sales
6	Reverse charge purchases

### SYSTaxPeriodStatusType

SYSTaxPeriodStatusTypeID	Name
0	Closed
1	Open
2	Deferred

### SYSTaxTranType

SYSTaxTranTypeID	Name
0	Sales Invoice
1	Sales Credit Note
2	Purchase Invoice
3	Purchase Credit Note
4	Sales Tax Journal
5	Purchase Tax Journal

### SYSTraderGenerateReasonType

SYSTraderGenerateReasonTypeId	Name
0	Unknown
1	Finance Charge Interest
2	Bad Debt Processing

### SYSTraderLocationType

SYSTraderLocationTypeId	Name
0	Main
1	Invoice
2	Delivery

### SYSTraderRevalAllocType

SYSTraderRevalAllocTypeId	Name
0	Allocation
1	Revaluation

### SYSTraderTranType

SYSTraderTranTypeId	Name
0	Deleted record
1	Purchase Payment Sales Receipt
2	Purchase Receipt Sales Payment
3	(not specified)
4	Invoice
5	Credit Note
6	Opening Balance Invoice
7	Opening Balance Credit Note

### SYSTraderWriteOffEntryType

SYSTraderWriteOffEntryTypeId	Name
0	Not Applicable
1	Write off Entry
2	Write off Account
3	Bad Debt

### SYSTranDateValidationClass

SYSTranDateValidationClassId	Name
0	Normal
1	Acceptable
2	Unacceptable

### SYSTranDateValidationType

SYSTranDateValidationTypeId	Name
0	None
1	Period
2	Calendar

## TermsOfDelivery

TermsOfDeliveryId	Code	CountryCode
0	EXW	GB
1	FCA	GB
2	FAS	GB
3	FOB	GB
4	CFR	GB
5	CIF	GB
6	CPT	GB
7	CIP	GB
8	DAF	GB
9	DES	GB
10	DEQ	GB
11	DDU	GB
12	DDP	GB
13	XXX	GB

## TimeUnit

TimeUnitId	TimeUnitName	MultipleOfDays
0	Days	1
1	Weeks	7
2	Months	30
3	Years	365

## TraceableItemStatus

TraceableItemStatusId	Name
0	Assigned
1	Unassigned
2	Prereceipt
3	Reserved
4	Direct Delivery

## TraceableTransType

TraceableTransTypeId	Name
0	Goods In
1	Goods In Return
2	Goods Out
3	Goods Out Return
4	Allocation
5	Allocation Adjustment
6	Allocation At Despatch
7	Transfer Out
8	Transfer In
9	Amendments
10	POP At Invoice
11	POP At Credit
12	BOM Goods Out

## TraceableType

TraceableTypeId	Name
0	Non-traceable
1	Serial Numbered
2	Batch Numbered

## TransactionGroup

TransactionGroupId	TransactionGroupName
0	Allocation
1	Goods Out
2	Goods In
3	On POP Order

## TransactionType

TransactionTypeId	TransactionTypeName	TransactionGroupId
0	Stock Addition	2
1	Internal Return	2
2	Return From Supplier	2
3	Stocktake Adjustment In	2
4	Transfer In	2
5	Write Off	1
6	Internal issue	1
7	Customer FOC Issue	1
8	Return To Supplier	1
9	Transfer Out	1
10	Stocktake Adjustment Out	1
11	Internal Allocation	0



12	Customer Allocation	0
13	Adjustment to Allocation	0
14	Opening Balance	2
15	Sales Order	1
16	Sales Return	2
17	Purchase Order	2
18	Purchase Return	1
19	POP Goods Received	2
20	POP Goods Returned	1
21	On POP Order	3
22	ON POP Order Adjustment	3
23	BOM/WO Issue	1
24	BOM/WO Build	2
25	BOM/WO Allocation	0

### TSActivitySearchKey

TSActivitySearchKeyID	Description
0	Code
1	Title

### TSCategoryCompStatusType

TSCategoryCompStatusTypeID	Description
0	Active
1	Inactive

### TSCategoryCompType

<b>TSCategoryCompTypeID</b>	<b>Description</b>
0	Group
1	Item

### TSClaimRecordStatusType

<b>TSClaimRecordStatusTypeID</b>	<b>Description</b>
0	Draft
1	Saved
2	Submitted
3	Rejected
4	Authorised
5	Posted
6	Complete

### TSConfigurableTerm

<b>TSConfigurableTermID</b>	<b>ConfiguredTerm</b>	<b>DefaultTerm</b>
0	Hierarchy	Hierarchy
1	Works Number	Works Number
2	Authoriser	Authoriser
3	Resource	Resource
4	Timesheet	Timesheet
5	Expense Claim	Expense Claim

### TSDayOfWeek

<b>TSDayOfWeekID</b>	<b>Description</b>
0	Sunday
1	Monday
2	Tuesday
3	Wednesday
4	Thursday
5	Friday
6	Saturday

### TExpenseClaimStatusType

<b>TExpenseClaimStatusTypeID</b>	<b>Description</b>
0	Draft
1	Saved
2	Submitted
3	Rejected
4	Authorised
5	Posted
6	Posted

### TExpensesPostingStatusType

<b>TExpensesPostingStatusTypeID</b>	<b>Description</b>
0	No
1	Yes
2	Required

### TSPaymentFrequencyType

TSPaymentFrequencyTypeID	Description
0	Weekly
1	Fortnightly
2	FourWeekly
3	Monthly
4	BiMonthly

### TSPaymentMethod

TSPaymentMethodID	Description
0	None
1	Cashbook
2	BACS
3	Payroll
4	SupplierAccount

### TSPostingMethodType

TSPostingMethodTypeID	Description
0	Transaction
1	PeriodBatch
2	SingleBatch

### TSReceiptAttachedType

TSReceiptAttachedTypeID	Description
0	Yes
1	No
2	N/A

### TSResourceHierarchyItemType

TSResourceHierarchyItemTypeID	Description
0	Group
1	Item

### TSResourceStatusType

TSResourceStatusTypeID	Description
0	Current
1	IsLeaving
2	SynchroniseWithPayroll

### TSResourceType

TSResourceTypeID	Description
0	Person

### TSSubmissionFrequencyType

TSSubmissionFrequencyTypeID	Description
0	Daily
1	Weekly

### TSTimesheetPostStatusType

TSTimesheetPostStatusTypeID	Description
0	No
1	Yes
2	Required

### TSTimesheetRecordStatusType

TSTimesheetRecordStatusTypeID	Description
0	Draft
1	Saved
2	Submitted
3	Authorised
4	Rejected
5	Complete

### TSTimeUnitType

TSTimeUnitTypeID	Description
0	Minutes
1	HoursAndMins
2	HoursDecimal
3	Days

### TSUserSearchKey

TSUserSearchKeyID	Description
0	Reference Number
1	Surname

### TSUserStatus

TSUserStatusID	Name
0	Inactive
1	Active

## TSWorkPeriod

<b>TSWorkPeriodID</b>	<b>Name</b>
0	Day
1	Week

## UnitOfMeasureType

<b>UnitOfMeasureTypeid</b>	<b>UnitOfmeasureTypeName</b>
0	Additional
1	Base
2	Stock

## WarehouseType

<b>WarehouseTypeID</b>	<b>WarehouseTypeName</b>
0	Normal
1	Home
2	Returns
3	Direct Delivery

## WopComponentLineType

<b>WopComponentLineTypeid</b>	<b>Name</b>
0	Component
1	Subassembly
2	Comment
3	Document
4	Platform

## WopOrderType

WopOrderTypeid	Name
0	BillOfMaterials
1	WorksOrder