

V1.0 16/06/2005. Initial version

EN570	New Returns Module enhancements General v5.70 enhancements
-------	---

*EL: Include in debug and production versions of defovr.inc, when resynch*

Purchase Return (Form 64)

## Record Mappings

?RN Line	Note	Type	Idetail Rec
Qty Expected	Place holder/ notation only	Real	QtyPick
Returned Qty	Qty sent/ received under cover of returns note. Also reduces live stock on PRN	Real	Qty
Replaced Qty	Qty of stock physically received/ sent out to cover returns note.	Real	QtyDel
Write off qty	Qty of stock written off as unserviceable	Real	QtyWOff
Book Stock Qty	Physical qty affecting live stock levels. Will mirror either Qty for PRN's, or QtyDel for SRN's	Real	Deduct Qty
Control Qty Replaced/issued	Keep so that any edits of Qty can be tracked	Real	QtyPWOff

Control Qty WOff	Keep so that any edits of Qty can be tracked	Real	SSDUplift
Serial Qty Returned	Track how many have been replaced /issued	Double	SerialRetQty
Bin Qty Returned	Track how many have been replaced /issued	Double	BinRetQty
Action Mode	Temp storage to control which qty controls action	Byte	JapDedType
Back to Back repair flag	Set when PRN created from SRN and needs processing through	Boolean	SSDUseLine
Warranty Line	Line was within warranty so whatever action is generated must be zero price.	Byte. 1 = Warranty line. 0 = Normal.	NOMIOFlg
Original trans G/L code	Copy of originating transaction G/L code. Used subsequently on any action transactions	LongInt	B2BLineNo
COS Original G/L code	Copy of COS G/L code likely to be used when Sales invoice posted.	LongInt	COSNomCode
Reason Code	Linked to reason for return	Byte	DocLTLink

?RN Header	Note	Type	InvRec
Status	Return Status	Byte	TransMode
Total qty noted	Qty expected back, sum of QtyPick	Real	DocLSplit[2]
Total Returned Qty	Sum of Qty	Real	DocLSplit[1]
Total Issued Qty	Sum of QtyDel	Real	DocLSplit[3]
Total Woff Qty	Sum of QtyWOff	Real	DocLSplit[4]
Total Amount issued	Total of QtyDel x Netvalue	Real	DocLSplit[6]

**New Passwords added**

Sales - Generate Sales Return 019 505  
Purchase - Generate Purchase Return 028 506  
Sales Orders - Generate Sales Return 035 507  
Purchase Orders - Generate Purchase Return 036 508  
Customer Details - View Returns 042 509  
Supplier Details - View Returns 043 510  
Stock Ledger - Generate Return 041 511  
Stock Serial/Batch - Generate Purchase Return 039 512  
Stock Serial/Batch – Generate Sales Return 039 513

Works Orders - Generate Purchase Return 152 514

Returns – Allow Access to Return Line Reasons List 175 545

Sales Return - Access to Daybook 175 515  
Sales Return - Access to History Daybook 175 516

Sales Return - Add a Transaction 176 517  
Sales Return - Edit a Transaction 176 518  
Sales Return - Print a Transaction 176 519  
Sales Return - Find a Transaction 176 520  
Sales Return - Hold a Transaction 176 521  
Sales Return - Copy a Transaction 176 522  
Sales Return - Action an Individual Return 176 523  
Sales Return - Prompt to action when storing 176 524  
Sales Return - Action all Returns 176 525  
Sales Return - Re-Print Action Run 176 526  
Sales Return - Generate Purchase Return 176 527  
Sales Return – Allow Daybook Post 176 540

Sales Return - Access to Reports 177 528  
Sales Return - Allow Generation of Sales Accrual 183 543

Purchase Return - Access to Daybook 179 529  
Purchase Return - Access to History Daybook 179 530

Purchase Return - Add a Transaction 180 531  
Purchase Return - Edit a Transaction 180 532  
Purchase Return - Print a Transaction 180 533  
Purchase Return - Find a Transaction 180 534  
Purchase Return - Hold a Transaction 180 535  
Purchase Return - Copy a Transaction 180 536  
Purchase Return - Action an Individual Return 180 537  
Purchase Return - Prompt to action when storing 180 538  
Purchase Return - Action all Returns 180 539  
Purchase Return - Allow Daybook Post 180, 544

Purchase Return - Access to Reports 181 541

Purchase Return - Allow Generation of Purchase Accrual 182 542

## Note to Development.

Additional notes to development:

### Notes re set-up program and licensing:-

Module release code added for Returns Module. Mode 19?

### Hooks

The Return structure follows the Transaction and Transaction Line convention.

The following hooks have been implemented:-

Within main Return body form (Standard hook points taken from other transaction types):-

2000, 01 : Pre store  
2000, 06 : Pre Select Account field  
2000, 08 : Exit of Period  
2000, 09 : Allows override of Delivery address  
2000, 61-64 : UD Header fields.  
2000, 65 : Entry of YourRef  
2000, 82 : Completion validation  
2000, 170: Post Store  
2000,180: Validation of YourRef as part of UD Hook.  
2000,189 : Post Select Account field

Within Return Line

4000, 01 : Upon Entry of Qty Returned.  
4000, 03 : Upon Exit of Qty Returned.  
4000, 04 : Upon Exit of Discount  
4000, 05 : Upon Exit of Unit Price  
4000, 06 : Exit of VAT code field  
4000, 07 : Form Close  
4000, 08 : Line Form Active  
4000, 09 : Upon Entry of Unit Price  
4000, 11 : Stock Code exit - After price calculated, allows unit price to be altered.  
4000, 12 : Qty exit - After price calculated.  
4000, 20 : Line completion validation  
4000, 26 : Called each time line total calculated mainly for support of VAT calculations  
4000, 31-34 Line UD fields

### New fields

#### Idetail Record

```
{ $IFDEF EN570 }
  { 589 } SerialRetQty
           : Double;      { Amount returned to stock via serial nos }
  { 597 } BinRetQty: Double; { " " into multi bins }
  { 605 } Spare2    : Array[1..22] of Byte;
```

**StockRec Record**

```

{$IFDEF EN570}
    {988}    SWarranty :    Byte;                {Standard warranty count in
months or years}
    {989}    SWarrantyType
                :    Byte;                {Standard warranty Type 0
months or 1 years}
    {990}    MWarranty :    Byte;                {Manufacturers warranty
count in months or years}
    {991}    MWarrantyType
                :    Byte;                {Manufacturers warranty
Type 0 months or 1 years}
    {992}    QtyPReturn :    Double;            {Qty Returned on PRN}
    {1000}   ReturnGL  :    LongInt;           {G/L code used for G/L
movement}
    {1004}   ReStockPcnt:    Double;            {ReStock pcnt auto applied
as an additional cost}
    {1012}   ReStockGL  :    LongInt;           {ReStock G/L code on auto
line}
    {1016}   BOMDedComp :    Boolean;           {Override behaviour of
deduct comp if no stock}
    {1017}   PReturnGL  :    LongInt;           {Keep G/L code seperate for
Purchase return levels}
    {1021}   Spare2     :    Array[1..30] of Byte; {!!}

```

**Serial Record. SerialType= Record**

```

{$IFDEF EN570}
    {267}   ReturnSNo  :    Boolean; { * Flagged as came back via returns
module *}
    {268}   BatchRetQty:    Double; { * Amount of batch child returned *}
    {277}   RetDoc     :    String[10]; { Sales Return Reference for match
}
    {287}   RetDocLine :    LongInt; { Line No. of Return }
    {291}   Spare      :    Array[1..64] of Byte;

```

**Location Master Record. MLocLocType = Record**

```

{$IFDEF EN570}
    {551}   loReturnGL :    LongInt; {Location Return GL}
    {555}   loPReturnGL:    LongInt; {Location Purchase Return}
    {559}   Spare2     :    Array[1..207] of Byte; {!!}

```

**Location Stock Record. MStkLocType = Record**

```

{$IFDEF EN570}
    {491}   lsSWarranty :    Byte;                {Standard
warranty count in months or years}
    {492}   lsSWarrantyType
                :    Byte;                {Standard warranty
Type 0 months or 1 years}
    {493}   lsMWarranty :    Byte;                {Manufacturers
warranty count in months or years}
    {494}   lsMWarrantyType
                :    Byte;                {Manufacturers
warranty Type 0 months or 1 years}
    {495}   lsQtyPReturn :    Double;            {Qty Returned on
PRN}
    {503}   lsReturnGL  :    LongInt;           {G/L code used
for Return G/L movement}
    {507}   lsReStockPcnt:    Double;            {ReStock pcnt
auto applied as an additional cost}
    {515}   lsReStockGL :    LongInt;           {ReStock G/L code
on auto line}
    {519}   lsBOMDedComp :    Boolean;           {Override
behaviour of deduct comp if no stock}
    {520}   lsQtyReturn :    Double;            {Sales Return
Qty}
    {528}   lsPReturnGL :    LongInt;           {G/L code used
for Purchase Return G/L movement}
    {532}   Spare2     :    Array[1..239] of Byte; {!!}

```

```

Multi Bin Record brBinRecType= Record
{$IFDEF EN570}
    {327} brReturnBin : Boolean; { * Flagged as came back via returns
module *}
    {328} Spare      : Array[1..438] of Byte;

```

```

Misc File = Record

```

```

    rtLReasonType = Record
        {002} CustomKey : String[20];      { ReasonCount padded to 4}
        {022} Spare1   : Array[1..6] of Byte;
        {029} UserKey  : String[20];      { Desc }
        {050} Spare3K  : String[10];      { Spare K }
        {061} ReasonDesc: String[60];      {Reason text}
        {121} ReasonCount: LongInt;        {Reason Order}
        {125} Spare    : Array[1..126] of Byte;

    end;

```

```

RecMFix = RetRLCode (Const 'R' in VarComnU.pas)
SubType = '#1'

```