

Ostendo[®]

Training

Copyright 2022 Development-X Limited

4th February 2022

Table of Contents

Introduction	11
1. Getting Started	12
1 Creating your TRAINING Database	12
2 Introduction to Flexibility and Options	13
Function keys	13
Quick Launch Toolbar	13
'List' Panel Viewing Options	14
Report Output Options	16
Analysis Views Options	16
System Alerts	20
Custom Menu Links	20
Custom Scripts	21
Data Import	22
Data Export	23
Custom Reporting to Spreadsheet	24
Inbuilt Report Writer	24
Global Name Change	25
Global Data merge	25
User Defined Fields	26
Company Required Fields	26
Auto Complete	27
Specific Email Text	27
Specific Form Layouts	27
System Screen Conditions	28
Desktop Views	28
Workflows	29
3 Function's Setup and Options	29
User Setup and Options	29
Administrator Functions	32
Financial Setup and Control Functions	32
Integration with your Accounting system	34
2. Items Descriptors and Labour Codes	37
1 Items	37
2 Descriptors	50
3 Labour Codes and Employees	54
3. Inventory Control	57
1 Warehouses and Locations	57
2 Inventory Movements	58
3 Inventory Replenishment	61

4 Create Required Orders	65
5 Order Inventory Availability	66
6 Inventory Changes	69
7 Assembly Order Backflushing	70
4. Sell Price Buy Price and Costs	73
1 Sell Price Maintenance and Use	73
2 Buy Price maintenance and use	79
3 Costing	82
5. Customers Sales Orders, Rentals and Invoicing	94
1 Customers	94
2 The Sales Order Process	97
3 Creating and Monitoring a Quotation	103
4 Creating and Monitoring a Sales Order	106
5 Creating a One-Step Order - no Backorder	115
6 Creating a One-Step Order with Backorders	117
7 Creating and Monitoring a Rental Order	119
6. Assembly Orders	122
1 Preparation	122
2 Bill Of Material	122
3 BOM Costs	125
4 By Products and Co Products	126
5 Assembly Orders	127
6 Labour	129
7 Assembly Order Backflushing	130
8 Assembly Order Closure	132
9 Custom Products	132
7. Job Orders	135
1 Customers	135
2 Preparation	138
3 Functions used in Quotations and Job Orders	139
4 Invoicing Options	145
5 Creating and Monitoring a Quotation	151
6 Receiving an Item from a Job Order	153
7 Jobs and Task Names	154
8 Jobs and Task Bills	155
9 Jobs and Templates	156

10	Costs and Values	158
11	Projects, Costs and Values	159
12	Using the Job Calendar	159
8.	Service Orders	163
1	Preparation	163
2	Customer Asset creation	164
3	Other Asset information	165
4	Customer Asset Service Scheduling	168
5	Breakdown / Repair Jobs	170
6	Warranties	170
9.	Suppliers and Purchase Orders	171
1	Preparation	171
2	Create Supplier records	172
3	Supplier Catalogues	174
4	Supplier Prices	175
5	Purchase Orders	178
6	Purchase Order Receipts	182
7	Purchase Order Invoices	183
8	Overseas Suppliers	185
9	Shipments	186
10.	CRM	188
1	Preparation	188
2	Contacts	188
3	Call Centre	189
4	Service Level Agreements	191
5	Activity Events	194
11.	Point of Sale	197
1	Preparation	197
2	POS - Retail Counter Sale	200
3	POS - Retail Pickup Order	202
4	POS - Retail Delivery Order	202
5	POS - Workshop Order	203
6	POS - Layby Order	204
7	Gift Vouchers	205
8	Cash Drops	206
9	End of Day	207
10	Other Functions	207

12. Constraint Based Scheduling	210
1 Terminology	210
2 Brief Overview of the Process	210
3 The First Exercise	211
4 Capacity Types	214
5 Scheduling Options	215
6 Planned -v- Firm Schedule	218
13. Workflow Graphical Designer	219
1 Getting Started	219
2 Graphics Tools	219
Straight Line	219
Rectangle	220
Ellipse	221
Text	221
Picture	222
3 Curve Edit Tools	222
Creating a Triangle	223
Creating a Curve	223
Curve Manipulation	224
More Curve Manipulation	225
4 Layout Tools	226
Viewing priority	226
Transparency	226
Object Grouping	227
5 Zoom Tools	227
6 Grid Tools	228
7 Alignment Tools	228
8 File Tools	229
9 Translation Tools	229
10 Inspector	229
Recap	230
Hint	230
Name	231
11 Layers and Schemes	231
Layers	231
Schemes	232
12 Library	233
Creating a new Library	233
Adding Objects to a Library	233
Adding Library Items to the canvas	234
13 Simple Examples	234
Oval Button	234
Corner Pipe	235
Electric Symbol	236

14	Creating a Process Flow	236
	Creating the Process	236
	Display on Ostendo Desktop	237
	Using the Process Flow	238
	Adding a KPI	238
	Link an Object to your website	238
14.	Report and View Developer	240
1	Reports	240
	The Report	240
	Defining the Report.....	240
	Amending the Report.....	243
	Adding Detail Lines.....	247
	Optional Printing.....	250
	Report Totals.....	252
	Reports -v- Forms.....	254
	User-Defined Parameters.....	255
	Print Detail.....	255
	Number of Printed Copies.....	257
	Other Report Features.....	258
	Freespace.....	258
	Using the Two-Pass Option.....	259
	Updating Ostendo Database.....	263
	Select Report Formats.....	264
	User-Defined Variables.....	267
	Run Ostendo Action from Preview.....	269
	Restrict Printing to Specific Users.....	270
	Other Report actions.....	270
	Creating a Data Field.....	270
	Changing a Data Field.....	271
	Creating a Text Field.....	272
	Adding a Date to a Report.....	273
	Adding Page Numbers.....	275
	Adding a Report Title.....	277
	Generating Sub Totals.....	279
	Creating Conditional Highlighting.....	281
	Adding Barcodes.....	282
	Adding Lines and Shapes.....	283
	Creating a Multiple Column Report.....	284
	Adding a Record Count.....	284
	Printing part of a Field.....	285
	Printing a Linked Image.....	287
	Printing Items with linked Properties.....	289
	Selective printing within the report.....	290
2	Analysis Views	291
	Creating an Analysis View	291
	Working the Analysis View	292
	Merging data with Microsoft Word	294
3	Chart Views	296
	Creating a Chart View	296
	Viewing the Chart	297
4	Pivot Views	299

Creating a Pivot View	299
Viewing the Pivot View	300
5 Inquiry Screens	303
Creating an Inquiry Screen	303
Working the Inquiry Screen	305
Adding a 'drill-down' Inquiry	306
15. Beginners Guide to Scripting	308
1 The First Script	308
2 Basic Structure	308
Introduction	309
Identifiers	309
Character 'String'	310
Comments	310
Variables and Constants	310
3 Standard Routines	311
Begin - End Constructor	311
Procedure and Function Declaration	311
If - Then - Else	312
For - To - Do	312
Arrays	313
Case Statements	313
4 Pre-defined Ostendo Functions	314
User Entered Data	314
AskQuestion.....	314
AskQuestionWithLookup.....	315
AskMandatoryQuestion.....	316
AskMandatoryQuestionWithLookup.....	316
AskQuestionNumericRange.....	316
AskQuestionWithUserDefinedLookup.....	317
DisplayData.....	317
Data from Ostendo Database	318
GetBooleanFromTable.....	318
GetStringFromTable.....	319
GetIntegerFromTable.....	319
GetDateFromTable.....	320
GetDoubleFromTable.....	320
GetCost.....	321
GetStdBuyPrice.....	321
GetStdSellPrice.....	322
GetSQLResult.....	322
Data from a Spreadsheet	323
LoadSpreadsheet.....	323
SSGetCellText.....	323
SSGetColumnCount.....	324
SSGetRowCount.....	324
Import a single Cell and Update an Ostendo record.....	324
Import and create multiple Ostendo records.....	325
Processing Functions	325
Execute SQL.....	325
InsertRecord.....	326
Progress Bar.....	327

Run	328
SetScreenParameter.....	328
RunSystemAction.....	328
Run Standard Ostendo routines via Scripting	329
Create a Job Order.....	329
InsertJob.....	329
InsertJobOrderLine.....	330
Create a Sales Order.....	330
InsertSalesOrder.....	330
InsertSalesOrderLine.....	331
Create a Purchase Order.....	331
InsertPurchaseOrder.....	331
InsertPurchaseOrderLine.....	332
Create an Assembly Order.....	332
InsertAssembly.....	333
InsertAssemblyStep.....	333
InsertAssemblyLine.....	334
InsertAssemblyOutput.....	334
Create a Timesheet Batch.....	335
InsertTimesheetHeader.....	335
InsertTimesheetLine.....	336
Inventory Replenishment Run.....	337
Run an Ostendo Report.....	337
Run an Ostendo Analysis View.....	338
Run an Ostendo Chart View.....	338
Run an Ostendo Pivot View.....	339
Run an Ostendo Inquiry.....	339
5 Where Scripts are Used	340
Main Menu Script	340
Desktop Icon Script	341
Command Line Script	341
Related Menu Script	342
Order Script	343
Screen Data Script	344
Custom Product Scripts	345
InsertBOMHeader.....	346
InsertBOMStep.....	346
InsertBOMLine.....	346
InsertBOMProperty.....	346
InsertBOMResource.....	347
SetBOMInstructions.....	347
SetBOMLeadTime.....	347
SetBOMRunDuration.....	347
SetBOMSetupDuration.....	347
GetCustomerSellPrice.....	348
A Custom Product Script Example.....	348
Workflow Script	349
SetWorkflowObjectColour.....	350
SetWorkflowObjectGradientColour.....	350
SetWorkflowObjectHint.....	350
SetWorkflowObjectText.....	351
SetWorkflowObjectTransparency.....	351
SetWorkflowObjectVisible.....	351
WorkflowObjectLoadPicture.....	351

Report Layout Editor	352
16. Custom Screens	353
1 Custom Data Screens	353
Creating your first Data Screen Script	354
The Data Entry Screen - Step 1	354
The Data Entry Screen - Step 2	356
Closing the Screen	359
Data Validation	360
Updating Ostendo	362
Recap	369
Other Functions	370
ProcessBarcode	372
DataScreenGetObjectText	373
DataScreenObjectLoadPicture	374
DataScreenSaveGraphicalFile	376
DataScreenSetEditText	377
DataScreenSetObjectColour	378
DataScreenSetObjectGradientColour	379
DataScreenSetObjectHint	380
DataScreenSetObjectTransparency	381
DataScreenSetObjectVisible	382
DataScreenChangeScheme	383
DataScreenActiveScheme	384
Simple Example	386
Advanced Example	387
2 Custom Data Entry	395
Data Entry Form	395
Creating the Input Panel	396
Defining the fields	397
Other Data Entry Functions	398
DataEntrySetLabel	398
DataEntryNewRecordValuesSet	399
DataEntryCellValueGet	399
DataEntryCellValueSet	400
DataEntryColumnCount	401
DataEntryRecordCount	401
Data Entry Procedures	402
DataEntryFocusedItemChanged	402
17. User-Defined Tables	404
1 Preparation	404
2 Creating the Tables in Ostendo	404
3 Generating the Edit View	405
4 The Script	406
5 Reports and Views	409
6 User-Defined Menus	413
7 Show Help	413
18. Accounting Interface	415

1 MYOB	415
Overview	415
Setup	416
Create and post your first Journal	419
Cost Centres	422
Base Mapped Cost Centres	425
'T' Charts	427
Advanced Cost Centre Mapping	434
2 Quickbooks	436
Overview	437
Setup	437
Create and post your first Journal	440
Cost Centres	443
Base Mapped Cost Centres	446
'T' Charts	447
Advanced Cost Centre Mapping	455
3 Sage Pastel Evolution	457
Overview	458
Setup	458
Create and post your first Journal	461
Cost Centres	464
Base Mapped Cost Centres	467
'T' Charts	468
Advanced Cost Centre Mapping	476

1 Introduction

The following Training Exercises take you through the various aspects of Ostendo using the 'DEMO' database. It is not the objective here to look at all the available options but to provide you with an understanding of how the various sections function both independently and integrated across Ostendo

2 1. Getting Started

These 'Getting Started' exercises take you through the following areas of Ostendo:

- Initial creation and Setup of your Training Database
- Introduction to flexibility and options available in Ostendo
- User-specific display and Inquiry options
- Administrators system setup and other functions
- Financial setup functions

2.1 Creating your TRAINING Database

Click on the '**Demo Ostendo**' icon on your desktop to go into the '**Demo**' database. Sign On as **ADMIN** with password **pass** (Note: The password is case sensitive)

Go to **File>System Configuration>Copy>Create Company** and ensure that the '**Copy**' Radio Button is selected. Now:

- From the drop-down list under '**Copy From Company**' select '**DEMO**'.
- Enter new company '**Training**' in the field '**New Company ID**'
- In the field '**New Database File Name**' click on the 'three dots' icon and point to where the database will be located. Ostendo gives the new database a name of '**Ostendo.fdb**' but you can change this to another identifier if you wish
- Having identified the new database and it's location click the '**OK**' button and Ostendo will generate the Training database

2.2. Log into the Training Database

Go to **File>Change Company** and select the '**Training**' database. Once again sign in as **ADMIN** with password **pass**

2.3. System Settings

Go to **File>System Configuration>System Settings** and set up your Training Company Information in the '**Company Information**' tab. In this screen there are a couple of Tables that are referenced. These already contain basic information but you can go into them and add more if required. The Tables are:

Default Company Site: By default Ostendo will use the Address that you entered in the System Settings screen and is the Site to where Purchased Goods are to be received. If you receive goods into a different location then go into **General>Company Sites** and add that site address then return and select it here.

Units of Weight, Length and Volume: These fields define the basic units of Weight, Length, and Volume that are used across Ostendo. The base system comes with Metric Units as the default. If you use (say) Ounces, Pounds, Feet, Miles, Square Feet, etc then you should first create them via **General>Settings>Standard Units** then return and them here.

2.2 Introduction to Flexibility and Options

2.2.1 Function keys

The following Function Keys are used throughout Ostendo. You don't need to try these out but be aware that they are available and will speed up data enquiry and entry processes.

F1 - Help

F2 - Where a 'Notes' field has limited visible space for data entry, clicking on this button will open a separate screen for entry of the extended notes

F3 - This functions as a 'Look-Up' shortcut in the following instances

- For fields with a Look-Up (spyglass) Icon this will display the table.
- For 'Notes' fields the 'Frequently Used Text' screen will be presented for selection of a Text/Phrase.

F4 - When cursor in field this enables drop-down lists to be displayed

F5 - 'Add' a new record. On multi-line entry screens this button will save the current line and automatically move the cursor to a new line (I.e. a combined F6 and F5 function)

F6 - 'Save' the current record

F7 - 'Cancel' information keyed in since accessing the record or 'Save' was selected

F8 - 'Delete' the current record

The following are also available

Shift/F5 - When in Job Orders, Sales Orders, Assembly Orders, Purchase Orders and Direct Invoicing screens this will automatically save the current Order details and bring up a panel for creation of a new order

Ctrl/Tab - Allows you to move from left to right across a suite of screens (For example: if the current view contains List, Detail, Lines, and Variants tabs then Ctrl Tab takes you forward through these tabs

Shift/Ctrl/Tab - Allows you to move from right to left across a suite of screens (For example: if the current view contains List, Detail, Lines, and Variants tabs then Ctrl Tab takes you backwards through these tabs

Tab - Move from field to the next field

Shift/Tab - Move backwards from field to the previous field

Space Bar - 'Check' or 'uncheck' a checkbox

Home - Go to first record in a 'List' View

End - Go to last record in a 'List' View

Page Up - Display previous 'Page' of data in a 'List' View

Page Down - Display next 'Page' of data in a 'List' View

2.2.2 Quick Launch Toolbar

To help go directly to common functions a configurable 'User-Specific' Quick Launch Toolbar is available.

You will see that the 'Quick Launch' Toolbar is located under the top toolbar. If you click on the extreme right of this Toolbar then a drop-down list of Ostendo's major functions are presented. Select one or two and make them visible on the Quick Launch bar

You can also click on the extreme left of the Quick Launch bar and drag the bar to any location on your screen.

2.2.3 'List' Panel Viewing Options

All of the 'List' panels throughout Ostendo have some - or all - of the following to provide the best possible viewing option to suit the end user. To see how this works go into [Sales>Customers](#) and remain in the 'List' tab.

1. Displayed fields

The displayed fields are taken from the master record to which the List applies (in this instance it is the Customer Master file). The selection of which fields to display is at your discretion. To select fields simply 'right mouse' in the main panel and select '[Customize List Fields](#)'. (If this option is not displayed then you don't have the rights to customise the screen as held against the User record - see [File>System Configuration>User Security and Options](#)).

On the displayed panel you can:

- Click on the '[Show field](#)' checkbox to display the field
- Amend the column heading by changing the content of '[Display Label](#)'
- Define the sort sequence of the displayed records by going to the lower panel and dragging the field from '[Available Fields](#)' to '[Sort By](#)'.
- Click the '[Save](#)' Button when done.

Try adding field '[Customer Region Code](#)' to the 'List' view

2. Field Position

On each List screen you can move the field position by dragging the column heading left or right to the position where you want it to appear. Click on the Column Heading of the '[Customer Region Code](#)' that you have just created and drag it to the left of '[Address 1](#)' field

3. Filtering and Sorting

If you 'check' the '[Filtering and Sorting](#)' checkbox at the bottom of the screen then the displayed data is available for filtering and sorting:

- To sort the data in a selected column simply click on the column heading. Clicking against will sort in descending order.
- Click on the black 'down arrow' to the right of your selected column heading and you can either select the specific entries to display records containing that entry, or you can select '[Custom...](#)' to enter detailed selection criteria.

4. Searching

Many options are available for you to search the displayed records. If you enter some text into the '[Search](#)' field and click on the 'binoculars' Icon then all records will be interrogated and those records that contain the entered text in the displayed fields will be displayed. If you enter multiple words enclosed in "double quotes" then the search will look for the combined words as they appear within the quotation marks.

You should also note that a multi-word search option can be carried out to your system preference. This can be either '[And](#)' to search for both words or '[Or](#)' to search for either word. To set the preferred option you should go into [File>System Configuration>System Settings](#) and 'check' the '[And](#)' or '[Or](#)' radio button

While you are still in [File>System Configuration>System Settings](#) 'check' the 'Advanced Searching' option. This gives you the option to also search other data linked to the current prime record. For example, In the Item 'List' screen ([Inventory>Items](#)):

Enter a Barcode ID to view which Item(s) are referenced to it
Enter Customer Part Number to show the Customer(s) and your own Item Number
Enter Supplier Part Number to show the Supplier(s) and your own Item Number
Enter an Item Property to view all Items that use that Property

Other display options are available depending upon the screen you are in (Example: in the Customer List screen ([Sales>Customers](#)) you can key in a Contact Name and all Customers that have a Contact with that name will be displayed

5. Other Viewing Options

If you right-click on any column 'Heading' a panel will appear that offers the following options that you should try. Note: Some options may be 'greyed out' because the above '[Filtering and Sorting](#)' checkbox has not been 'checked'.

Sort Ascending: The current column will be sorted into ascending sequence

Sort Descending: The current column will be sorted into descending sequence

Clear Sort: This will clear the sort sequence in the current column

Group By This Field: A '[band](#)' will appear above the list into which the current field now resides as a 'box'. The data in the main panel is grouped by this selection. If you click on the '+' sign in the main panel then the group will expand to show all records in the group. Note that:

- You are not restricted to a single grouping in the top Group Box. I.e. You can have Groups within a Group.
- You can also 'change' the grouping sequence by simply 'dragging' one of the Group Boxes left or right as required
- If you 'right mouse' on a Group Box you will note that the option 'Group by this field' now reads 'Remove from Grouping'. Select this to remove this Group Box and restore the details to the main panel

Group By Box: A '[band](#)' will appear above the list. You can select and drag any column heading into this area where it will now reside as a 'box'.

- If you 'right mouse' on a Group Box you will note that the option 'Group by this field' now reads 'Remove from Grouping'. Select this to remove this Group Box and restore the details to the main panel.
- If you select the 'Group Box' again then it will not appear in the panel; however any Groupings will still be retained

Footer: A band will appear at the bottom of the display. Under each column - if you click the 'right mouse' - you have the option to display:

- Sum - The total of the values in the column (only if it is numeric)
- Max - The maximum value that exists in the column (only if it is numeric)
- Min - The minimum value that exists in the column (only if it is numeric)
- Ave - The Average of all the values in the column (only if it is numeric)
- Count - The Number of records in the column
- None - No value displayed

Group Footers: Similar to the above 'Footer' but the values appear at each 'Group' level

Remove this column: This will 'hide' the column from the current display. To restore the column select '[Field Chooser](#)'

Field Chooser: The fields displayed in the main panel are defined in the above 'Displayed

Fields' option. Using this feature you can 'Hide' selected fields from those on display. When '**Field Chooser**' is selected a small panel will appear. You can hide fields by clicking on the column header and dragging it to this panel. Restoring the field to the display is simply the reverse of this action

Best Fit: Selecting this will adjust the width of the selected column to that defined by the maximum width of the data entered into it, or the Column Title, whichever is the greater

Best Fit (All Columns): Performs the same function as 'Best Fit' on all the columns in the display

2.2.4 Report Output Options

Whenever a report is run you are presented with a panel that gives you three options available:

- **Screen** - View on Screen
- **Print** - Send to Printer
- **Email** - Email with this report attached as a pdf file.

To see this in action minimise the current Customer screen and go to **Sales>Reports** and select one of the Customer Listing reports. You can ignore the entry parameters and go straight to the Report

2.2.5 Analysis Views Options

'**Views**' in Ostendo provides facility for a User to define their own view of data in the Ostendo database, analyse that information, and output to various media. Whenever a '**View**' is run the relevant information is returned from which you can carry out the following functions. Go into **Sales>Views>Analysis - Customers** and carry out some of these options

Search & Sort: The following features are available that allow you to sort and filter the displayed data.

- Sort any column in the displayed records into ascending or descending sequence by clicking on the selected column heading
- You may 'filter' the records based upon your own selection criteria by clicking on the blue triangular symbol in the selected column heading. From the drop-down list you may select the following:-
 - All - displays all records
 - Blanks - Displays only those records that have no data in the field
 - Non-Blanks - Displays only those records that contain data in this field
 - Select the specific field content

Moving and Hiding Columns: You may also move columns as well as take unwanted columns from the display

- You can move columns by clicking on the column heading and 'dragging' the column into the required position.
 - If you double click on the Column Heading's right edge you can make the column automatically 'close up' to match the amount of data in the field. You can also achieve this by 'Right Clicking' the column heading and selecting 'Best Fit'.
 - If you wish to 'Close Up' all the columns then you should 'Right Click' on any column heading and select 'Best Fit - All Columns'.
 - If you 'Right Click' on a column Heading and select 'Field Chooser' then a panel will appear for storing unwanted columns and enables you to 'Customise' the screen. You can move any unwanted columns to this panel by simply clicking on the column heading and dragging the column into this panel. You can recall stored columns by reversing this procedure.
-

An alternative to this is to 'Right Click' on a column Heading and select 'Remove this column'. This will automatically place the unwanted column in the storage panel

To hide the 'Field Chooser' Panel simply click on the 'x' in the upper right corner of the panel

Grouping: You may wish to Group 'like' records. To do this simply drag the required column heading into the area at the top of the screen where it states 'Drag a column header here to group by that column'. (If this area is not visible then you should 'Right Click' on any column heading and select 'Group By Box')

The screen will now group all records where the content of the 'Grouped' field is the same.

- If you click on the '+' indicator against each Group you can see the detailed records.
- This Grouping facility is not just single level. You can Group within Group, etc by simply dragging and dropping the 'sub-group' to the right of the first Group.
- This can be repeated for as many levels as you require.
- An alternative method of achieving this is to 'Right Click' on the selected column heading and select 'Group By This Field')

Summary Totals: For each Group you may wish to display summary totals. To do this, expand a Group by clicking on the '+' indicator to display the individual records. At the bottom of the Group List is a blank area. If this area is not visible then 'Right Click' on a column heading and select 'Group Footers'.

Now go to this blank area under any column and 'right click' the mouse to display the following options. These can be accessed depending upon the type of field (For example, you cannot 'Sum' a Date field)

- | | |
|-----------|------------------------------|
| ▪ Sum | Numeric fields only |
| ▪ Min | Numeric and Date fields only |
| ▪ Max | Numeric and Date fields only |
| ▪ Count | All fields |
| ▪ Average | Numeric fields only |
| ▪ None | All fields |

The selected column will now display the summary information

Grand Totals: You may also wish to display grand totals for the displayed data. To do this you should see a blank area at the end of the displayed list. If this area is not visible then 'Right Click' on any column Heading and select 'Footer'. As with Group Footers, you can go to the blank area under any column and 'right click' the mouse to display the options.

Mail Merging

You have the option to print a document from within Ostendo whereby 'User-defined' source data can be merged with a Word document to produce printed forms with data from both sources

In this exercise we will create a letter to be sent to certain Suppliers. The Letter contains a combination of data fields from Ostendo and user Text. In our example we will produce the following letter

To <Company Name>
<Address Line 1>
<Address Line 2>
<Address Line 3>

Dear <Primary Contact>

This is my test document for merging.

Regards

Step 1. Extract the Supplier Data

Go into **File>Reporting Configuration>Report and View Developer**. Click the 'Add' button and, in the creation screen, 'check' the 'Analysis' Radio Button and enter 'Suppliers' in the 'Name' field

On the presented 'Master Settings' tab:

- Select 'Purchasing' from the drop-down against **Category**
- Against 'Merge Word Document' enter a document name (Example: **SuppLetter**) that will become the Merge Template Document. Note: use the 3 dots to generate the **full** path
- Against 'Merge Data File' enter a file name (Example: **SuppSource**) that will become the Merge Source Data. Note: use the 3 dots to generate the **full** path

Now enter a query in 'Master Query' that will extract the data you wish to use. For example

```
Select SUPPLIER, SUPPLIERADDRESS1, SUPPLIERADDRESS2,  
SUPPLIERADDRESS3, PRIMARYCONTACT from SUPPLIERMASTER
```

Also enter a Supplier From and To parameter as follows

```
From Supplier;SUPPLIERMASTER.SUPPLIER >= :FSUPPLIER  
From Supplier;SUPPLIERMASTER.SUPPLIER <= :TSUPPLIER  
Both linked to Code Type 1001
```

'Save' the entries

Step 2. Run the Analysis and create the Data Source

This step will create the extracted data format from which the Merge Template can be created.

Click the 'Preview' button to produce an Analysis View of the extracted data.

Click on 'Export' on the top toolbar and select 'Mail Merge'

This routine will generate your **SuppSource** data file containing the data displayed in the Analysis View.

The first time through you will also get a message stating '**File Not Found. Make sure the file path is correct**'. This is because the Word document to print the information has not been created yet. That is the next step.

Step 3. Create the Word Document and link to the Data Source

This step will create the Document using Microsoft Word and link to the generated fields in the extracted **SuppSource** file.

Go into Microsoft Word and create a new document and 'Save' it as **SuppLetter** in the path that you identified above.

Click on 'Tools' on the top toolbar within Word and select 'Mail Merge'. A panel will appear into which you should carry out the following steps.

Step 3.1. Click the 'Create' button and select 'Form Letters'

Step 3.2. On the displayed panel select 'Active Window'

Click on 'Get Data' button and then on the 'Open Data Source' option and point it the 'SuppSource' document that you generated above. A panel will appear asking you to define what the Field and Record delimiters are. Select the following

Field Delimiter - , (i.e. A comma)

Record Delimiter - (enter)

Click the 'OK' button

Step 3.3. On the presented panel click the 'Edit Main Document' button.

You have now linked the document to the data source. The next step is to select the data fields and place them on the document.

Step 4. Constructing the Merge Document

This step will create the Document using Microsoft Word and link to the generated fields in

In the new document that you have just generated you will notice that a new toolbar called 'Mail Merge' appears at the top of the screen. In that Toolbar click on the 'Insert Merge Field' and you will see the field names in the 'SuppSource' file.

Type in the word 'To' followed by a space on the first line. The position of the cursor now defined where the first 'merged field' will appear.

From the 'Insert Merge Field' select field 'SUPPLIER'. Your document should now look like this

To << SUPPLIER>>

Complete the remainder of the document using a combination of typed words and linked Data Fields to produce the following

To <<SUPPLIER>>
<<SUPPLIERADDRESS1>>
<<SUPPLIERADDRESS2>>
<<SUPPLIERADDRESS3>>

Dear <<PRIMARYCONTACT>>

This is my test document for merging.

Regards

then 'Save' the document

Step 5. Preview

You can preview the finished report by clicking on the '<<ABC>>' button on the 'Mail Merge' Toolbar and, using the right and left arrows on the same Toolbar, progress through all the extracted records. You have now created the 'SuppLetter' document to use in conjunction with the 'SuppSource' file. Close all documents and we will now see the full process flow

Step 6. Running the Merged document from within Ostendo

Go into Ostendo and select 'Purchasing>Views' then select 'Suppliers'.

Click on 'Export' on the top toolbar and select 'Mail Merge' and you will be presented with the Merge document as you created above.

If you click on the print icon you will print the design view that you currently see

If you click on **File>Print** you will print the merged details
You can also click on the **'Merge'** button on the Merge Toolbar and print the merged details

2.2.6 System Alerts

System Alerts (can be seen and allocated via **File>System Configuration>System Alerts**) provide facility to have a 'pop-up' message appear on your PC at a selected time after the alert condition has occurred. (For Example: A Purchase Order has overdue Lines). The notification can then be forwarded (emailed) to the relevant Employee should any action be required.

Whenever an Alert is 'triggered' a notification appears on the screen for a few seconds.

To see alerts in action go into **File>System Configuration>System Alerts** and go down to Alert Module **'Inventory'** and Alert Name **'Inventory Below Minimum'**. Against this record you can see the following:

Alert Active: If this is 'checked' then Ostendo will monitor the Alert

Assigned To: You can annotate which User(s) get this Alert. Currently it will alert you because you are signed in as **'ADMIN'**

Alert Severity: You can set your own severity level. See later for how this is displayed

Therefore this Alert will be triggered whenever the stock of an item goes below its Re-Order Level. We will use Item **110-2041** (Washer-Stainless Steel-16mm). If you go into **Inventory>Items** and select Item **110-2041** you will see that it currently has stock of **2000** and a Re-Order Level of **2000**. If we issue some of this then the current stock will fall below the Re-order Level and the Alert will be triggered. Therefore, go into **Inventory>Inventory Adjustments** and click the **'Add'** button. **'Save'** the batch then click on the **'Lines'** tab. Click on the **'Add'** button and - in the **Item Code** field - select **110-2041** and enter **-1** in the **Adjustment Qty +/-** field. **'Save'** the record then go back to the **'Detail'** tab and click on the **'Post all Adjustments'** button. Upon posting the batch you will see the Alert for a few seconds. You can view all your Alerts by clicking on the **'View Alerts'** button. If the small Alert panel disappears you can still view the Alerts by clicking on one of three 'icons' displayed on the bottom right of the screen (**'Low'**, **'Medium'**, or **'High'** severity) and the Alerts for that severity is displayed. Within that screen you can display Alerts of other severity levels by 'checking' the checkbox(es) at the top of the panel

2.2.7 Custom Menu Links

You can create links to other programs, files or functions in your computer network and have them displayed under their own menu within Ostendo. Selecting the entry will automatically activate the program, file or function.

Go to **File>Custom Menu Links** and click the **'Add'** button. Enter the name of a program that you may use from within Ostendo. For example **'Excel'** and enter the full path of the program's executable in the **'Filename'** field (For example: **C:\Program Files\Microsoft Office\Office\Excel.exe**) then **'Save'** the entry.

You will now see a new entry **'Custom'** on the top toolbar of Ostendo. If you select this you will see **'Excel'**. Clicking on this will immediately start the program.

2.2.8 Custom Scripts

For the more advanced User, Pascal Scripts can be created to perform a multitude of user-defined tasks. Before we go into the various ways that Scripts can be used let's do a simple example

Go into **File>Custom Menu Scripts** and click the **Add** button. Enter a **Script Name** of (say) **Message** then enter the following between the **Begin** and **End** text.

```
showmessage ('This is my test of Custom Scripts');
```

Save the record and exit **Custom Menu Scripts**. If you now click on **Custom** on the top toolbar you will see **Message**. Clicking on this will action the Script.

The above example gives a simple demonstration of Script. Now try creating another Custom Script called (say) **Cust Copy** using the following

```
executesql('update customermaster set CUSTOMERNOTES = Substring(Customer from  
1 for 10)');  
showmessage ('Customers Copied');
```

If you run the script then all Customer Records will have the first 10 characters of the Customer Name copied to the **Notes** field of each Customer record. To see the results of running the script go into **Sales>Customers** and select the **Detail** tab against any Customer. You will see the first 10 characters in the **Notes** field at the bottom of the screen

Types of Scripts

Having seen how a Script works let's look in more detail at the ways in which we can use Scripts. It is not the objective here to take you through each option in detail but merely to identify the options available.

1. 'Custom' Menu Script

This is the basic Script style that adds the Script Name to the drop-down under 'Custom' on Ostendo's top Toolbar. Selecting this will run the Script. This style of Script can be:

- Available to all
- Restricted to Administrator
- Restricted to specific Users

Examples of the types of script are:

- Updating your existing system with information from Ostendo
- Updating Ostendo with information from your existing system
- Importing Supplier Catalogues
- Send user-defined KPI information via email, or to a Mobile phone

2. 'Related' Menu Script

This adds the Script Name to the drop-down Menu held against the 'Related' button within specific screens. Examples could be as above with information specifically related to the current main screen

3. Order Script

This allows you to create a Script that is run against an Assembly, Job, Sales, Purchase or POS Order and enables you to add extra specific functionality such as:

- Total Order Value discounting based on Order content
- Freight calculations based on Order content

- Order Authorisation Levels (Example: User Purchasing levels)
- Order Margin Control with User-defined Margin levels
- Order Validation and/or Checks
- Workflow actions (Example: send Email regarding this Order)
- Promotions (Example: 3 for price of 2, etc)

The Script can also be defined as 'Mandatory' in which case Picking, Receiving, Printing, etc is denied until the Script has been run

4. Screen Data Script

This is related to Master Order, Receiving, and Invoicing Screens where the action of Adding or Deleting a record or changing any field within the record will automatically run the Script to provide a resultant action. For example

- Zero Price Check on Sales Order Lines
- Update Sell Price based on Last receipt Cost
- In Purchasing check for best price from all Suppliers
- Have a pop-up Sales Message appear
- Specify a minimum order quantity
- Show active Promotion when Sales Line Entered

5. Accounting Link Script

This type of script enables you to define links to other Financial Accounting Packages. You can define scripts that will output in the appropriate format to suit the standards required by the receiving Accounting system

2.2.9 Data Import

You can import any table directly into Ostendo. This will involve matching your data fields to required Ostendo fields. Data integrity checks will be carried out during the import.

In our exercise let us import a couple of **Pricing Groups** into the Pricing Group Table found under. If you go to **Pricing>Settings>Pricing Groups** you will see that currently there are no records

To prepare for importing create and save a spreadsheet with the following entries (including the column headings).

Name	Description
Retail	All Retail Price Levels
Wholesale	All Wholesale Price Levels

Within Ostendo go to **File>Data Importing** and go through the following steps

Step 1: Click on 'Radio' button Excel Spreadsheet (.xls) file format then click the 'Next' button.

Step 2: Select '**PRICINGGROUPS**' (I.e. The Table into which the data is being imported) and click the 'Next' button

Step 3: Point to where you saved the above spreadsheet and then enter the following:

Start at Row - Enter **2** because row 1 contains the column headings

This Row contains Field Names - Enter **1**

Click the 'Next' button.

Step 4: You should now match the fields in your .xls file with the equivalent fields required in Ostendo. (Note: You do not need to match the SysUniqueID because Ostendo will allocate this

when the record is created). Once each field has been mapped click the **'Next'** button.

Step 5: Define the format of the fields in the Import File. This allows the import routine to convert it from the .xls file format to the format defined in the Regional Settings of your computer. As you have just created the .xls file then no changes will be required therefore click the **'Next'** button.

Step 6: You can now tell the Import routine what you want it to do with the data relative to existing information in Ostendo. I.e.:

- **Append** - Add all records even if they currently exist in Ostendo
- **Update** - Update existing Ostendo records where there is a matching key. You will be required to identify the unique key field identity from the drop-down list
- **Append/Update** - Add the record if it doesn't exist and update the record where there is a matching key. You will be required to identify the unique key field identity from the drop-down list.
- **Delete** - Delete existing records where there is a matching key. You will be required to identify the unique key field identity from the drop-down list
- **Copy** - Delete all existing records in Ostendo and recreate from the import file.
- **Append New** - Add a new record where the record in the Import file does not currently exist in Ostendo. You will be required to identify the unique key field identity from the drop-down list

This screen defaults to **'Append'** which is the option that you are going to use. Therefore just click the **'Next'** button.

Step 7: This screen confirms your settings. Ensure that all the settings are correct and then click on the **'Execute'** button to import the records.

If you now close the Import Routine and go back to **Pricing>Settings>Pricing Groups** you will see that the records have been added.

2.2.10 Data Export

This function allows you to export data from Ostendo to either a Comma Separated Value file or an Excel Spreadsheet. The procedure uses a simple Wizard that takes you through the whole process. In this exercise we will export selected fields from the Item Master table

Go to **File>Data Exporting** and go through the following steps

Step 1: Click on the Excel Spreadsheet (.xls) **'Radio'** button then click the **'Next'** button.

Step 2: Point to where the generated file will reside and give it a name then click the **'Next'** button.

Step 3: Select **'ITEMMASTER'** and click the **'Next'** button

Step 4: All the fields in the **'ITEMMASTER'** table will be displayed. They are all pre-selected for output. You have the option to:

- Select specific fields by checking or unchecking the checkbox adjacent to each field
- You can alter the sequence in which they are output by dragging and dropping the field into the required position

Select the fields that you want to export then click the **'Next'** button.

Step 5: Define the format of the fields in the Exported File if the format is to be different to that set up in the Regional Settings of your computer. Click the **'Next'** button

Step 6: This screen confirms your settings. Ensure that all the settings are correct then click the **'Next'** button

Click the **Execute** button to perform the export.

Have a look at the exported spreadsheet where you will see the details from your Item Master table

2.2.11 Custom Reporting to Spreadsheet

This feature opens up with a blank spreadsheet into which you can extract data held within Ostendo. Go into **General>Data Spreadsheet** and you will be presented with a blank spreadsheet. You have a couple of options:

1. You can open an existing Spreadsheet by selecting **File** then **Open Spreadsheet** and selecting (say) the spreadsheet that you generated during the above **Import** routine
2. You can extract information from data within Ostendo using a Query. You will find instructions on SQL Query under **Help>Reporting** on Ostendo's Toolbar. For this exercise, however, we will do a simple extract from the Customer Master file.

In the **General>Data Spreadsheet** screen click on **Data** then **Query Builder**. In the presented screen type **'C'** to take you down the available Tables and then find **CUSTOMERMASTER**. Double click on this and the following will appear in the **'Query'** panel

Select from CUSTOMERMASTER

And the cursor will be positioned between **Select** and **from**. Type in an asterisk and the line should now be

Select * from CUSTOMERMASTER

(This statement 'says' extract all fields from the Customer Master table)

Click on the **'Run'** button (don't save the query) and you will see all the Customer data appearing in the Spreadsheet. The extracted information can be saved and/or printed as required.

2.2.12 Inbuilt Report Writer

The term 'Report' refers to:

Report: Provides standard user-defined reports from Ostendo data

Views: Allows the User to view, Group, Total and analyse information.

Pivot Charts: Drag and Drop columns and rows with summaries and drill-downs

Charts View: A pictorial representation of various statistics with drill down options

All 'Reports' supplied with Ostendo can be copied then amended and made specific to the current signed on Company. I.e. You cannot change a basic Ostendo Report but you can copy a Report to your Company Reports Folder and make amendments to the copy. When printing a Report Ostendo will always look into your Company Reports folder first and then into the base folder if a Company specific report doesn't exist.

Let us create a simple Report called (say) **'Item Listing'**. Therefore go into **File>Report Configuration>Report and View Developer** and click the **'Add'** button. On the displayed panel 'check' the **'Report'** Radio Button and enter Name **'Item Listing'** then click the **'Create'** button.

In the upper part of the screen we need to define where the Report should appear Ostendo's Menu structure. Therefore complete the following fields

- **Include in Main Menu:** 'Check' this so that the Report will 'Reports' section under Inventory
- **Name:** Leave this as **'Item Listing'**
- **Menu Order:** Enter a sequence number that defines the position of Report menu
- **Category:** Select **'Inventory'** from the drop-down list so that the finished Report will appear under **Inventory>Reports**
- **Type:** Leave this as **'Report'**
- **Specific Screen:** Leave Blank
- **Report Filename:** Leave this as **'Item Listing'**

Master SQL: In the area under 'Master Query' enter the following:

```
Select * from ITEMMASTER
```

Note: You could have clicked on the **'SQL Builder'** button and generated it in a similar manner to creating the Query in 3.11. (above).

Having defined the data selection criteria you now need to create the actual Report. Therefore click on the **'Edit'** button to the right of the screen and select **'Standard Report Wizard'** in the presented screen. On the wizard panel click on the **'Fields'** tab and move (say) the following fields to the **'Selected Fields'** area

```
ITEMCODE
ITEMDESCRIPTION
ITEMUNIT
ITEMSTATUS
```

Have a look at the **'Layout'** and **'Style'** tabs and make your selection before clicking the **'Finish'** button. You have now created the Report. To see the finished result click on the **'Preview'** Icon at the top-left of the screen (or key **Ctrl+P**).

Save the Report and exit the Report and View Developer. If you now go to **Inventory>Reports** you will see that **'Item Listing'** is in the list. Click on that to print the Report.

2.2.13 Global Name Change

There are instances when you may wish to globally change the identity of (for example) a Customer, or Inventory Item. Let us assume that Customer **'Seven Wonders Ltd'** is now **'Worldwide Exports'**. Go into **File>System Configuration>Global Name Change** and select **'CUSTOMERMASTER-CUSTOMER'** in the first field. In the next field select **'Seven Wonders Ltd'** and, finally, in the third field enter **'Worldwide Exports'**. Click on the **'Apply change to entire database'** button.

If you now go into **Sales>Customers** then you will see that the Customer Name has been changed along with all current Orders and History records.

2.2.14 Global Data merge

Whereas the 'Global Name Change' allows you to rename existing fields to a totally new name this feature allows you to rename an exiting field to one that that already exists in Ostendo. Thus effectively merging two fields. Go into **General>Settings>Departments** where you see the current Departments in the system. Let us say that Department **'Service'** has now merged with Department **'Sales'** and we need to change all references in Ostendo from one to the other. If you also go to **Labour>Employees** you will see that **'Terry Jones'** is linked to the **'Service'**

Department

Go into **File>System Configuration>Global Data Merge** and select '**DEPARTMENTS-DEPARTMENTCODE**' in the first field. In the next field select '**Service**' and, finally, in the third field enter '**Sales**'. Click on the '**Merge all records from Old to New Value**' button.

If you go back to **General>Settings>Departments** where you see that Departments in the system. You will see that Department '**Service**' does not now exist and that if you go to **Labour>Employees** you will see that '**Terry Jones**' is linked to the '**Sales**' Department

2.2.15 User Defined Fields

Numerous key areas of Ostendo have additional fields against which you can utilise with your own field definitions. These fields will be immediately available for data entry and maintenance screens in addition to reporting and analysis views.

In this exercise we will add a new field against Items called '**MilSpec**' (Military Specification) and allow Item enquiries to be made by MilSpec.

Go to **File>Customisation Configuration>Additional Fields** and click the '**Add**' button. Enter the following in the created record

Module: Select '**Items**'
Caption: Enter '**MilSpec**'
Field Type: Select '**Text**'
Value List: Leave blank

'**Save**' the record

If you now go to **Inventory>Items** and select (say) **1500-2188** (Yellow Paint) then click on the '**Detail**' tab you will see (half way down the left-hand side of the screen - a tab called '**Additional Fields**'). If you click on this you will see your created field into you which you can enter data. Enter (say) **MilSpec-38823**

Note: If you go back to the '**List**' view you can include this additional field in the view by 'Right Mousing' in the middle of the screen and selecting 'Customize List Field' then selecting '**ADDITIONALFIELD_1**' and entering a '**Display Label**' of '**MilSpec**'

You can now Filter, Search and Group by **MilSpec**

2.2.16 Company Required Fields

Ostendo has, by default, many fields that are mandatory during data entry. Other fields are optional. This feature allows you to define those 'optional' fields that are to be 'mandatory' for your Company. If, during data entry, this field is not completed then an error message will be returned showing a user-defined description of the omission.

To see how this works go into **File>System Configuration>Required Fields** and 'check' the '**Display System Required Fields**' checkbox. You will see all the '**System**' fields that have been defined as '**Mandatory**' by Development-X and these cannot be changed. 'Uncheck' this checkbox then let's make, for example, the '**MilSpec**' field (created in the previous step) a Mandatory requirement during data entry. Therefore click the '**Add**' button to add a new record

Table Name: Select '**ITEMMASTER**'
Field Name: Select '**ADDITIONALFIELD_1**'
Display Field Name: Enter '**Military Spec**'

'Save' the record

If you now go into **Inventory>Items** and try to add a New Item (or amend an existing Item) then you will get a message requiring an entry to be made in the **'Military Spec'** field

2.2.17 Auto Complete

If you go into **File>System Configuration>System Settings** then you will see an **'Enable Auto Complete on Look-up'** checkbox. If this is checked then - on any field in Ostendo that has a drop-down list from which to select an entry - you can begin typing characters and the drop-down display will commence at that sequence of characters. If the sequence of characters is unique to a specific record then the remaining characters will automatically populate the field.

You've already encountered an example of this when you first signed into Ostendo. I.e. You didn't type the full **ADMIN** but just **A** and - because there is only one User beginning with **A** - the remainder of the entry automatically populates the field

2.2.18 Specific Email Text

This feature allows you to create and maintain Email Text to accompany various Forms (Example: Invoice, Purchase Order, Quote, etc). This Text will populate an Email whenever the respective 'Form' is appended and emailed to a Customer or Supplier. The Text can contain plain text or can also include data taken from the attached Form. Let us create a specific email Text to accompany a Sales Invoice.

Go into **File>Reporting Configuration>Specific Email Text** and click the **'Add'** button. Under **'Print Form Name'** select **'Invoice'**.

Under **'Email Subject'** enter **Invoice ["INVOICENUMBER"]**

Under **'Email Body'** enter

To ["PRIMARYCONTACT"]

Please find attached our Invoice ["INVOICENUMBER"] dated ["INVOICEDATE"] for your attention

You will see that the Text is a combination of open-format Text and defined fields. The defined fields are those that are available under the 'Master Query' of the specific report that is attached. In this instance it will be all fields under MD_ within the Sales Invoice Report.

Go into **Sales>Customers** and click on the **'Detail'** tab against existing Customer **'Jim Gold & Co Ltd'**. Ensure that the Customer has both an **'Email'** address and a **'Primary Contact'**.

Now go to **Sales>Direct Invoicing** and create a new Invoice against **'Jim Gold & Co Ltd'**. Add a Line to the Invoice in the **'Lines'** tab and then return to the **'Detail'** tab and click on the **'Print'** Icon and then on the **'OK'** button in the presented panel. You will be shown the three options to output the Invoice. You should select the third option (**Email**). An email will be generated with the above information contained in the email and a pdf copy of the Invoice attached.

2.2.19 Specific Form Layouts

The feature allows you to link nominated Customers or Suppliers to specifically designed forms. Using the Inbuilt Report Writer you can copy and amend a standard report layout and **'Save as'** a different report name. This screen then enables you to establish a link between nominated Customers, Customer Types, Suppliers, etc. and this report. When running the report Ostendo will

use its layout in preference to the standard layout.

Create a copy of the Sales Invoice Form and name it (say) **SalesInvoice01.fr3**. Amend the copy by (for example) putting a different Company Logo on the form.

Go to **File>Reporting Configuration>Specific Form Layouts** and click the 'Add' button. In the new record enter the following

Print Form Name: Select 'Invoice'
Condition: Select 'Customer'
Value: Select 'Jim Gold & Co Ltd'.
Filename: Click on the 3 dots icon and locate **SalesInvoice01.fr3**

'Save' the record

If you now go to **Sales>Direct Invoicing** and create a new Invoice against 'Jim Gold & Co Ltd'. Add a Line to the Invoice in the 'Lines' tab and then return to the 'Detail' tab and click on the 'Print' Icon and then on the 'OK' button in the presented panel. You will be shown the three options to output the Invoice. You should select the first option (**Screen**) and you will see that the **SalesInvoice01.fr3** layout will be used

2.2.20 System Screen Conditions

This screen allows you to identify - by User - restrictions regarding which records can be addressed. When the User goes into the relevant screen(s) only those records defined here are available. In this example we will restrict the view of Suppliers to those Suppliers whose Supplier Type is 'General'

Go into **File>System Configuration>System Screen Conditions**. Click the 'Add' button to create a new record. Enter the following:

Select **User** 'ADMIN'
Select **Screen** 'Suppliers'
Filter Condition Enter **SUPPLIERTYPE = 'General'**

If you now go into **Purchasing>Suppliers** you will see that you are restricted to viewing 'General' Suppliers only.

Go back into **File>System Configuration>System Screen Conditions** and delete the restriction

2.2.21 Desktop Views

The feature allows each User to have up to 4 different views displayed on the Ostendo Desktop. These are views that you can create via Ostendo's Report and View Developer function. They can also be dynamic in that the view can be refreshed at regular intervals using current Ostendo Data. Examples of Views that could reside on the Desktop are:

- Bar chart of Sales By Salesperson
 - KPI's for various activities with drill-downs on detailed data
 - Pie-Chart of Sales by Region
 - etc
-

2.2.22 Workflows

The feature allows each User to be allocated a personalised Workflow, which is presented in a graphical format. This Workflow is displayed on their Ostendo Desktop. Clicking on an Object in the workflow will take them to a Screen or Report relating to that object. The setting up and use of this feature is demonstrated in a separate Exercise. (Ostendo Graphical Designer).

2.3 Function's Setup and Options

These Exercises show some of the functions that can be used to configure Ostendo to suit the Business requirements

2.3.1 User Setup and Options

You can set up Users in the following manner

Go into **File>System Configuration>User Security and Options** and click the **'Add'** button. Add your own name to the **'UserName'** field; complete the rest of the information and identify yourself as a **'Normal'** User.

1. User's Screen and Reports Access

Now let's have a look at what you can do with a **'Normal'** User. Note: an **'Administrator'** User has access to everything.

If you click on the **'Security Access'** tab you can select all Screens, Reports, and Views to which the User will be allowed access. For now click on the **'Clear All'** button and uncheck everything. Next go into **Inventory** and 'check' the first three in the list (Inventory, Items, and Descriptors) only. **'Save'** the changes.

Click on **File>Switch Users** and sign in with your new User Name and password **'pass'**. Having seen the effect you should, once again, click on **File>Switch Users** and sign in as **ADMIN** and go back to **File>System Configuration>User Security and Options**. Highlight your new User and click on the **'Security Access'** tab and click on the **'Check All'** button.

2. User's Screen Formatting Options

If you now click on the **'User Options'** tab to see what this offers. This first thing you will notice that these settings are specific to the User. You may wish to try each option out by enabling the option here and attempt to carry out the specific function

Save Grid Layouts: If this is 'checked' then any changes made to the Grid Layouts by this User will be saved when the User exits Ostendo. Those settings will remain until the User Changes them in the relevant screen or the **'Reset Grid Layouts'** button is selected.

Reset Grid Layouts: If this is selected then all Grid Layouts will revert to the Ostendo basic format.

List Customising: If this is 'checked' then option to **'Customise List Fields'** (this will allow the User to make changes to Lists views such as adding or removing columns or amending their sort sequence.

Reset List Grids: If this is selected then amended Customised List Grids will revert to the Ostendo basic format.

3. Approvals

Ostendo offers a feature whereby by certain transactions MUST be approved before its next step can be carried out. 'Approvals' are available against the following activities.

- Timesheets
- Purchase Receipts
- Purchase Invoices
- Assembly Issues
- Assembly Receipts
- Job Invoices
- Job Issues

Let us see how this works.

- In the '**User Options**' tab of the User Record that you created above 'uncheck' the '**Allow Approvals**' checkbox.
- Go to **Purchasing>Settings>Purchase Rules** and 'check' the '**Purchase Invoice Approvals**' checkbox. This denotes that Approvals are required before Purchase Invoices can be processed after initial entry.
- Now create a Purchase Invoice by going into **Purchasing>Purchase Invoicing** and select (say) Supplier '**Bruce Wilson**' and annotate the '**Invoice Style**' as '**Invoice Only**'
- Go to the '**Lines**' tab, click the '**Add**' button and add '**Item Code**' '**100-2002**' for quantity **1000**
- Go back to the '**Detail**' tab and you will see that the '**Approval Status**' is '**Waiting Approval**' and, because you don't have Approvals Authority, the '**Post Purchase Invoice**' button has not been activated and you will not be allowed to Post the Invoice
- Go back to the '**User Options**' tab in the User record and 'check' the '**Allow Approvals**' checkbox.
- Go back to the Purchase Invoice that you created above and you will now see that you can amend the '**Approval Status**'. If you change this to '**Approved**' you will notice that the '**Post Purchase Invoice**' button is now active and the Invoice can be Posted

4. Disable Price Override

Ostendo offers a feature whereby by Users can be prevented from over-riding system evaluated:

- Sell price against Sales Order Lines
- Sell Price against Job Order Lines

Let us see how this works.

- In the '**User Options**' tab of the User Record that you created above and 'uncheck' the '**Cannot Change Order Prices**' checkbox.
- Go to **Sales>Sales Orders** and select any Sales Order then click on the '**Lines**' tab.
- You will see that the '**Unit Price**' field is 'greyed out' and the Sell Price cannot be amended
- If you go back to the User record and 'uncheck' the '**Cannot Change Order Prices**' checkbox and then return to this Order you will see that you are now allowed to override the Price

5. Alert Screen not displayed on login

By default - and if you have any Alerts - you will always get the Alerts screen displaying when you first log into Ostendo. You can stop this by going into the '**User Options**' tab of the above User Record and 'uncheck' the '**Alert Screen is not displayed on Login**' checkbox.

When this is 'checked' the User will have to click on the Alert Icons at the bottom-right of the

screen to view current Alerts.

6. Purchase Order Alert Limit

A System Alert (**Purchase Limit Warning**) is activated whenever the User raises a Purchase order that exceeds a defined limit for that User. To demonstrate this:

- Go into the '**User Options**' tab of the above User Record and enter a '**Purchase Order Alert Limit**' of (say) **\$1000**
- Go into **File>System Configuration>System Alerts** and assign this to User **ADMIN**
- Go to **Purchasing>Purchase Orders** and create an order against (say) Supplier '**Bruce Wilson**'. Go to the '**Lines**' tab and add '**Item Code**' **1800-2190** (Rear Wheel Assembly) then enter an order quantity of **20** and '**Save**' the record
- Upon saving the record an Alert will be generated for the assigned 'User'. If you are already signed in as **ADMIN** this should appear at the bottom-right of the screen. If you are not currently signed in as **ADMIN** then go to **File>Change User** and sign in as **ADMIN**. You should see the Alert is now displayed for your perusal/action.

7. Desktop Views Panel

Go to the '**Desktop Views**' tab of the User Record that you created above. This panel allows you to select up to 4 Desktop Views that will appear on the Ostendo Desktop and lie on top of the basic Ostendo screen. The Views themselves should already have been created via the Report and Views Developer function (**File>System Configuration>Report and View Developer**). For the purpose of this exercise we will use a couple of existing views to demonstrate how to set up the Desktop for the current User.

Under the '**Top Left**' section enter the following

View Type: select '**Chart**'

View: select '**Chart - Open Sales Orders**'

'**Save**' the settings

Now go to **File>Switch User** and sign in as the above User. You should now see the Desktop.

8. Workflow Panel

Go to the '**Workflow**' tab of the User Record that you created above. This panel allows you to select a Workflow that will appear on the Ostendo Desktop and lie on top of the basic Ostendo screen. The Workflow can be designed specifically for the User if required. From within the Workflow you can 'Drill Down' to specific Screens. Let's see how this works

In this screen you should first 'check' the '**Enable Workflow**' checkbox and then select '**SampleWorkflow.dat**' from the drop-down list. (This is located directly under the Ostendo Folder). '**Save**' the selection and '**Close**' out of the User record. You should see the Workflow displayed on your desktop. (If you still have 'Desktop Views' enabled you will have to click on the 'Workflow' tab to see this).

In the Workflow click on the '**Customers**' Button and you will be taken directly to the 'Customer' screen of Ostendo.

A separate Tutorial (**Ostendo Graphical Designer**) is available that takes you through the Workflow Design process

2.3.2 Administrator Functions

These are the functions and features that are normally used by the Systems Administrator

1. Auto Numbering

You can define start numbers for the various Numbers used in Ostendo. To view the current numbers go into **File>System Configuration>System Settings** and click on the **'System Numbering'** tab. It is strongly recommended that you use numbers starting with (say) 100000 rather than 1. This is because computer number sorting would then be in a logical sequence. I.e.

100001
100002
....
100015
100016
etc

rather than

1
15
16
2

2. Copy / Create Company

You have already created a database 'by copy' in Exercise 2.1. This feature can be very useful if you are trying out new functionality on 'Live' data. You simply copy your 'Live' company and generate a 'test' company against which you will try out the new functions.

3. Multiple Companies

When signing into a Company you can instantly recognise that you are in the correct company by use of a colour code. Go into **File>System Configuration>Multiple Companies**. If you click on the drop-down against field **'Company Colour'** you can select the colour. This colour takes effect the next time you sign into the company

4. Active Connections

If you go to **File>Active Connections** and click on the **'Get Connection Details'** button Ostendo will respond by displaying the current signed-on Users. This is very useful in a Client/Server environment if you wish to stop the Server (for whatever reason) and you need to inform all current Ostendo Users to finish their current transactions

2.3.3 Financial Setup and Control Functions

These are the functions and features that can be used by the Financial Controller

1. Tax Groups, Codes, and Tax matrix

An individual line on an Invoice can have its own Tax Rate. Therefore Ostendo has been designed to accommodate this in the following manner

Go into **File>Financial Configuration>Tax Codes** and look at the current Codes already defined. In this exercise we will use the Tax Code **'Zero'**

Go into **File>Financial Configuration>Tax Groups** and look at the current Groups already defined. Add another Tax Group called (say) **'Finance'**

Go into **Inventory>Descriptors** and create a new Descriptor **'Legal Services'** (You also need a Unit of Measure (\$) and a short Description). Also in the **'Detail'** tab you will see a field called **'Tax Group'**. Select **'Finance'** from the drop-down list. Finally, add a Sell Price of **\$80** then **'Save'** the record

Go into **Sales>Customer** and look at the details against Customer **'Jim Gold & Co Ltd'**. You will see that the Customer has a Tax Group of **'TAXABLE'**

What we are now going to establish is that when **'Legal Services'** is sold to **'Jim Gold & Co Ltd'** then the chargeable Tax will be **'Zero'**

Go into **File>Financial Configuration>Tax Matrix** and add the following:

Customer/Supplier Tax Group: TAXABLE
Item/Descriptor Tax Group: Finance
Tax Code: Zero

If you go into **Sales>Sales Orders** and click on the **'Add'** button you can create a Sales Order for Customer **'Jim Gold & Co Ltd'**. Click on the **'Lines'** tab against the generated order and **'Add'** Descriptor **'Legal Services'**. You will see that the chargeable Tax is Zero. If you add (say) Item Code **100-2004** you will see that a Tax is charged for that Item because it's Tax Matrix states that **GST** Tax Rate is to be applied

2. Credit Terms

Go into **File>Financial Configuration>Credit Terms** and look at the Terms already defined. Add your own if you have Terms that differ from any of these.

If you now go into **Sales>Customer** and look at the details against Customer **'Jim Gold & Co Ltd'** you will see that this Customer has Terms taken from the Credit Terms Table. You can amend this here if required. You should also note that these Terms are copied to any generated Order where that can also be amended at the Order Level.

3. Cut-Off Dates

If you go into **File>Financial Configuration>Cutoff Dates** you will see three dates which allow you to maintain Financial Cut-Off Dates. These are used to prevent entry of transactions prior to that date. The transaction areas covered are:

- Orders, Order Transactions and Inventory
- Supplier Invoicing
- Customer Invoicing and Payments

If you also go into **File>System Configuration>System Settings** you will see a field called **'Max Entry Days Forward'**. The number of days entered in this field is added to the system date and prevents entry of transactions after the evaluated date.

4. Accounting Basis

If you go into **File>System Configuration>System Settings** you will see a field called **'Accounting Basis'**. From the drop-down select the basis by which your accounts are run. The options are:

Invoice: Sales Revenue and Tax amount are posted at time Invoice was raised

Cash Deferred Sales: Sales Revenue and Tax Amount are posted at time of payment.

Cash Immediate Sales: Sale Revenue is posted at time Invoice was raised. Tax Amount is

posted at time of Payment

5. Currencies

You should first go into **General>Cost Centres** and create a Cost Centre where the Value in that currency is stored. Create 3 Cost Centres - one for each of the following currencies - Euro, Pound Sterling and Yen

If you now go into **File>Financial Configuration>Currency Codes** you can create Foreign Currency Codes and their Exchange Rate. You should note that you do not need to enter your Home Currency as Ostendo will use the Regional Settings on your computer

For other Currencies enter the Code here. For the purpose of this exercise let's enter three currencies

- EUR Europe (Euro)
- GBP Great Britain (Pound)
- JPY Japan (Yen)

You should enter the 3-character code into field **Currency Code**

The Currency Symbol entered here is used across Ostendo and precedes the value where the currency is used (Example: £24.23, or €12.34, etc)

There are a couple of ways to get the Currency Symbol if it is not available on your keyboard

- Go into Microsoft Word and select Insert on the top toolbar. In the dropdown list select 'Symbol'. You can now select the symbol and 'Insert' into the word document. From there you can copy the symbol and paste it into this field
- Set your keyboard with 'Numerics Lock'. Using the enabled numbers due to the Numerics Lock being on (i.e. NOT the numbers 0 to 6 on the top row of the keyboard) you should place the cursor where the symbol is to appear and then - holding down the 'Alt' key - enter
0128 for the Euro (€)
0163 for the Pound (£)
0165 for the Yen (¥)

In the drop-down under **'Foreign Purchase receipts Cost Centre'** select the appropriate Cost Centre that you created above

Go into the **'Rates'** tab and select the Rate Type and the Exchange Rate. The Rate is entered as (for example) if this currency was GBP and your home currency was NZD and the exchange rate was 1 NZD = 0.3845 GBP then you should enter 0.3845

2.3.4 Integration with your Accounting system

These are the steps required in Ostendo to produce Journals for sending to your accounting system

1. Cost Centres

Ostendo uses the concept of Cost Centres to record costs from all areas of the product. Ostendo uses a separate program (Ostendo to Accounting System Link) in which these Cost Centres can be mapped to your Accounting System's Account Codes)

If you go to **General>Reports>Financial Flow - Sales** and run the report you will see the Journals that are created during the Sales Process. Focusing on the **'Sales Issue'** Journal you will see that it Debits **'Sales Lines Picked'** and Credits **'Items (Stock)'**

Go into **General>Cost Centres** where you will see some Cost Centres that are in the base Ostendo.

What is now required is to establish a link between **'Sales Lines Picked'** and Credits **'Items (Stock)'** and the Cost Centres that will provide the costs

2. Cost Centre Mapping

If you go into **File>Financial Configuration>Cost Centre Mapping** you will see that Ostendo already has these mapped out. For example: **'Items (Stock)'** is linked to Cost Centre **'STOCK'** and **'Sales Lines Picked'** is linked to Cost Centre **'SALES PICKED'**.

To see this in action go to **Sales>Sales Orders** and select Order **SO300013**. Click on the **'Lines'** tab and highlight Item **'ST-7622'** in the upper part of the screen and then click on the **'Picked Lines'** tab in the lower part. Click the **'Add'** button and enter a quantity of **1** then **'Save'** the record. Now let's have a look at the created Journal

Go into **General>Reports>Financial Batch Report** and enter **'From Date'** and **'To Date'** both equal to today's date. Click the **'OK'** button and you will see that a Journal has been created with the correct Cost Centres and Cost of the transaction

3. Sales Mapping

A Sales Mapping screen extends the above base functionality allowing you to establish a more detailed structure specifically covering the Sales area. It consists of a hierarchical structure using

- 1 - Invoice Customer Type
- 2 - Invoice Customer Region
- 3 - Order Customer Type
- 4 - Order Customer Region
- 5 - Order Class
- 6 - Order Type
- 7 - Sales Order Person
- 8 - Category

And mapping them to

- Income Cost Centre
- Cost of Goods Cost Centre

Let us say that we want to map all Customers in a Sales Region to specific Account Codes. To do this you should first go into **General>Cost Centres** and create a Cost Centre to represent the Income from this Region and another to represent the COGS from this Region

If you go to **Sales>Customers** you may see field **Customer Region Code** (Hint: If you don't see it then bring it onto the screen as described in 3.3.1.). You will see a Region Called **'Asia'**. We will map all Sales to 'Asia' to go to their own Accounts

Go to **File>Financial Configuration>Sales Mapping Matrix** and **'Add'** a record that contains the following

Invoice Customer Region: **Asia**

Income Cost Centre: Select the Cost Centre you created above

Cost of Goods Cost Centre: Select the Cost Centre you created above

Let's see this in action

Go to **Sales>Sales Orders** and create an Order for '**Golden Arch Trading Co**'. Go into the '**Lines**' tab and add **1** hour of '**GENERAL TIME**'. Go into the '**Picked Lines**' tab and issue the **1** hour. Finally, go back to the '**Detail**' tab and click the '**Sales Invoice**' button and print the Invoice.

If you now go into **General>Reports>Financial Batch Report** and enter '**From Date**' and '**To Date**' both equal to today's date. Click the '**OK**' button and you will see that Journals have been created with the above Cost Centres in the Journals

4. Labour Mapping

A Labour Mapping screen extends the above base functionality allowing you to establish a more detailed structure specifically covering the Labour area. It consists of a hierarchical structure using

- 1 - Category
- 2 - Labour Code Department
- 3 - Employee Department
- 4 - Labour Code
- 5 - Employee

And mapping them to

- Direct Labour Cost Centre
- Fixed OH Cost Centre
- Variable OH Cost Centre

The process is similar to the Sales Mapping

3 2. Items Descriptors and Labour Codes

This training session will show you the Items, Descriptors and Labour Code functions of Ostendo. Additionally, it will - using the Ostendo Demo Database - take you through an interactive session and allow you to carry out various activities.

3.1 Items

You should use Items if you either:

- Maintain Inventory, or
- Order Items for immediate use in an Order (i.e. by-pass Inventory)

You can view existing Items by going into [Inventory>Items](#). Select the 'Detail' tab if you wish to view Item Details. The following exercises will go through the various fields defined against an Item and, for each significant field, take you through a simple flow to demonstrate the use of that field

1. Setup

The Item record contains fields that are completed by making a selection from a user-defined list. You should have a look at these before creating your first Item record.

Units of Measure: Each Item record MUST be given a Unit of Measure. A short list of popular Units is included in the base Ostendo. Add your own Units of Measure by going into [General>Settings>Standard Units](#)

Tax Group: Each Item record MUST be given a Tax Group. This allows the combination of Item's Tax Group and Customer's (or Supplier's) Tax Group to denote the Tax Code (and therefore the Tax Rate) to be used on a Sales (or Purchase) Order Line. You can maintain Tax Groups via [File>Financial Configuration>Tax Groups](#). Within the Tax Group screen you can nominate a 'Default' Tax Group that will populate an Item record during Item creation. (You can, of course, amend the Tax Group in the created Item record)

Categories: This is used extensively for selection criteria when updating or Analysing data, printing reports, addressing Sales Financial Mapping. This is not essential and can be done at any time. To create and maintain Categories go into [Inventory>Settings>Categories](#)

Sub Categories: This enables sub-sections to the above Category to be defined. As with Categories it is used for selection criteria when updating or Analysing data, and printing reports. This is not essential and can be done at any time. To create and maintain Sub-Categories go into [Inventory>Settings>Sub-Categories](#)

Adjustment Types: Adjustment Types are used to identify the Type of stock movement being carried out. The base system already contains some Adjustment Types. If you don't wish to use these defaults then it is preferable to amend these before any transactions are carried out. To create or maintain Adjustment Types go to [Inventory>Settings>Adjustment Types](#)

Warehouses: Ostendo always uses a Warehouse in Inventory transactions. It can simply be a system-wide default Warehouse or multiple physical Warehouses from which a selection is made when carrying out a stock movement. The base Ostendo contains a Warehouse. This can be deleted if you have not yet created an Item that uses this. To add or maintain Warehouses go into [Inventory>Warehouses](#).

Locations: Ostendo always uses a Location within a Warehouse in Inventory transactions. It can simply be a system-wide default Warehouse/Location combination or multiple physical Locations

across multiple Warehouses. The base Ostendo contains a Warehouse/Location combination. This can be deleted if you have not yet created an Item. To add or maintain Locations and Warehouses/Location combinations go into **Inventory>Locations**.

Location Groups: If you are using Locations then you can (optionally) categorise them into groups (Example: Raw Materials, Oils, Hazardous Goods, etc). This facilitates reporting and analysis by these groups. To create or maintain Location Groups go into **Inventory>Settings>Location Groups**

Pricing Groups: Pricing Groups are used to facilitate 'Batch Update' of Sell prices using screen **Pricing>Batch Price Update**. If you will be using the Batch Price Update facility then create the Pricing Groups via **Pricing>Settings>Pricing Groups**. You can leave this for now and come back to it during the Price Update Exercises

Analysis Groups: Analysis Groups are used across Ostendo to facilitate viewing and reporting by Group. Additionally the Groups are used to format the print content within Quotes and Invoices. If you are going to use these features then you should first establish the Analysis Groups by going to **General>Settings>Analysis Groups**

Item Rules: These provide default settings that are used to prefill Item records during creation. You should go into **Inventory>Settings>Item Rules** to view the current settings or amend the Rules as required

2. Item Creation

Go into **Inventory>Items** and click on the 'Add' button then enter details to create your Item. The mandatory fields are the Item Code, Unit of Measure, and Description. Click the 'Save' button when done. Now let's have a look at some of the options surrounding Items.

2.1. Multiple Units of Measure

Ostendo allows you to define multiple Units of Measure against a single Item. Stock of the Item is segregated by UOM in addition to Sell Prices being associated with each UOM. In this exercise you will:

- Create an Item
- Link other Units of Measure (6-Pack, Dozen, Crate) to it
- Receive the Item in various Units of Measure
- View Stock that is segregated by Unit of Measure
- Split/Combine the current stock into other Units of Measure

Units of Measure are maintained via **General>Settings>Standard Units**. Go into this screen and add 'Bottle', '6-Pack', 'Dozen', and 'Crate'.

Now let's apply these to an Item. Go to **Inventory>Items** and click on the 'Add' button. Create a new Item called (say) 'Spring Water'. Select UOM of 'Bottle' from the drop-down list against field 'Unit'. Enter a description (Example: 'Simply Spring Water'). Enter a 'Std Sell Price' of (say) **\$1.50**. Click the 'Save' button to save the Item.

To add other Units of Measure to this Item you should click on the 'Related' Button found to the right of the screen and select 'Item Units' from the displayed list. You will see that the Base Unit ('Bottle') is already in this screen with a conversion factor of 1 and a Price Discount of 0. These cannot be amended.

Add a Unit of '6-Pack' with a Conversion Factor of 6 (I.e. there are 6 Bottles per 6-Pack) and a Sell Price Discount of 10% (I.e. the Sell Price of a 6-Pack is the sell Price of an single Bottle multiplied

by the Conversion Factor multiplied by (100% - Discount %)). Repeat this for the other Units ('**Dozen**', and '**Crate**').

To see the effect of these Units close the screen then go into **Inventory>Inventory Adjustments**. Add a new Batch and '**Save**'. Go into the '**Lines**' tab and create a new line. Go to the Item Code field and select '**Spring Water**' from the drop-down list (Hint: you can type 'S' and the cursor will immediately go to Items beginning with S). Enter an Adjustment Type of '**Receipt**' and quantity of **20** and click the '**Save**' button. Add another line for '**Spring Water**' but this time click on the '**Unit**' field and select another Unit of Measure. Repeat for the 3rd Unit of Measure. Finally go into the '**Detail**' tab and click on the '**Post all Adjustments**' button.

Go into **Inventory>Inventory Availability** screen, select '**Spring Water**' and then go to the '**Detail**' tab. You will now see the current stock of '**Spring Water**' in each Unit of Measure. While you are in this screen click on the '**Transaction History**' and '**Projected Availability**' tabs. Close the Inventory Availability screen.

If you wish to split or combine Units then go into **Inventory>Inventory Unit Change** and locate '**Spring Water**' then click the '**Details**' tab. Highlight the record that you wish to split/combine then click on the '**Change Unit of Measure for selected record**' Button. On the displayed panel enter the quantity being changed and the Unit Of Measure to which it is being changed then click the '**Update**' button. If you now go into **Inventory>Inventory Availability** screen you will see that the Unit Of Measure changes are recorded in the '**Transaction History**'.

2.2. Status

Ostendo has 4 statuses used by Items

Active - Is currently used throughout Ostendo

Planned - Can be Purchased only

Runout - Cannot be ordered but can be used until stock is zero

Obsolete - Cannot be ordered, Issued, or received

Change the Status of '**Spring Water**' to '**Planned**' then go into **Sales>Sales Orders**. Select Sales Order **SO300013** and click on the '**Lines**' tab. Add this Item to the Sales Order line and try to '**Save**' the line. You should get a message preventing you from adding this Item due to its '**Planned**' status. Cancel the line to exit the Sales Order.

As this Item will be used in later exercises reset its Status to '**Active**'

2.3. Default Supply Method

This defines whether the Item is normally supplied from stock or you should raise a supply order for each demand. The selection options are:

Supply From Stock - This means that issue to Sales Orders, Assembly Issues, Job Orders will come from stock. Items with this Supply Method will normally be maintained using an Inventory Replenishment routine.

Source On Demand - This means that whenever a demand (Sales Order, Assembly Issue, Job Order) is made then Ostendo will immediately create a matching 'Suggested' Supply Order (Assembly Order or Purchase order).

Use of these methods is covered in more detail under exercises covering '**Inventory Control**' Training Exercises.

2.4. Source By

In combination with the Default Supply Method this field defines from where the Item is procured:

The displayed options are:

Purchasing - This means that the Item will automatically be available to Purchasing whenever a requirement arises.

Assembling - This means that the Item will automatically be available to the 'Assembly' process whenever a requirement arises

Custom - This means that the Item will automatically using one of two methods. These are:

Use a configuration 'script' defined in the Assembly module of Ostendo

Use a pre-defined Bill of Material and amend during Order creation

The use of the **Purchasing** and **Assembling** options will be explained in more detail in the 'Inventory Control' Training Exercises. The use of the **Custom** option will be explained in more detail in the 'Assembly Orders' Training Exercises.

For now leave the 'Source By' set to **Purchasing**

2.5. Manufacturers Info

Many stockists, especially in the electronics industry, hold Manufacturers information against Items. This provides facility to enquire on Inventory by Manufacturer, Brand, or Model. In our exercise we will link **Spring Water** to a Manufacturer and Brand.

To prepare for this you should go into:

- **Inventory>Settings>Manufacturers** and create the Manufacturer's Names. Create an entry (say) **Waterworld**
- **Inventory>Settings>Brands** and create the Brand Names. Create an entry (say) **Pacific Blue**

Go into **Inventory>Items** and select **Spring Water** then click on the **Manufacturer's Info** tab and link the Manufacturer and Brand that you have just created to this item

Now let's view the Manufacturer and Brand on the Item **List** view. Go to the Item **List** tab 'right mouse' in the main panel and select **Customize List Fields**. (If this option is not displayed then you don't have the rights to customise the screen as held against the User record - see **File>System Configuration>User Security and Options**).

On the displayed panel you can:

- Locate field **MANUFACTURER** from the list
- Enter **Manufacturer** in **Display Label**
- Click on the **Show field** checkbox against this line
- Repeat for Field **BRANDNAME**
- Click the **Save** Button when done.

You will now see that the selected fields are visible in the **List** view. 'Check' the **Filtering and Sorting** checkbox at the bottom-left of the screen. Place your cursor to the column heading of **Manufacturer** and click on the black triangle. You can now restrict the display to a single Manufacturer. Click the cross at the bottom-left to restore the view

2.6. Sales Warranty

You can attach a Warranty Identity to an Item. The Warranty is carried through to Sales and Job Orders and becomes effective when the Item is picked

Go into **Service>Warranty Definitions** and view the Warranty Codes that are currently on file.

Add your own as required.

Go back to **Inventory>Items** for Item **'Spring Water'** and click on the **'Sales Warranty Applies'** checkbox. From the drop-down list select the applicable Warranty. Note: you can also maintain Warranty Codes by clicking on the icon to the right of the drop-down icon.

Go into **Sales>Sales Orders** and select Sales Order **SO300015**. On the Sales Order Lines add Item Code **'Spring Water'** and 'Save' the entry. If you now click on the **'Warranty'** tab you will see that the Warranty Code has been copied from the Item record to the Sales Order Line.

Whilst in the Sales Order Line screen click on the **'Picked Lines'** tab, click on the **'Add'** button and enter a quantity of (say) **1**. Click the **'Save'** button

Close the Sales Order screen then go into **Service>Warranty List** where you will see the generated Warranty record. You can, of course, amend the details as required.

2.7. Customer Assets

A Customer Asset can be physical Item or a general description of (for example) a building or collection to be regularly serviced. The Asset can be created as the result of a Sale of an Item or can be created directly in the Service Module. Each Asset can have a Service Schedule attached to it and, when the service is due, reminders sent to the Customer. This short exercise covers the creation of a Customer Asset as a result of the Sale of a product. The full Servicing and Maintenance routine is explained in detail in the **'Service Orders'** Training exercises.

Go to **Inventory>Items** and select Item **'AC-8026'**. In the Detail screen you will see that the field **'Create Customer Asset'** checkbox is 'checked'. An adjacent button (**Customer Asset Options**) will now become active. Clicking on this button will bring up a panel where you can define the Asset Type and (optionally) identify a Service Schedule (Asset Types and Service Schedules are user-maintained in the **'Service Orders'** Training Exercises and will not be covered here).

You should note that Items to be converted into Customer Assets must be **'Serial Controlled'**.

Go into **Sales>Sales Orders** and select Sales Order **SO300013**. Go to the Lines screen and add a new line to cover quantity **1** of **AC-8026**. Click on the **'Picked Lines'** tab and **'Add'** a new pick line. Place the cursor in the **'Qty'** field and do NOT enter a quantity. You should, however, click on the 'spyglass' icon to bring up a panel that shows the individual serial numbers in stock. Enter a quantity of **1** against a single Serial Number and **'Save'** the entry. Click the **'OK'** button to exit

If you now go to **Service>Customer Asset** you will see that the picked Item has now been created into an Asset against the Customer. (The Asset Identity will comprise of the Item Number followed by the Serial Number). Click on **'Details'** to view the originating Sales Order. Click on **'Planning and History'** to view the generated service schedule.

2.8. Add-On Sales

'Add-On' Sales uses a list of Items that you may also wish to promote during the Sales process when selling a specific Item. For example, If the Item was a tin of Paint then you may wish to suggest the purchase of Paint Brushes, Brush Cleaner, etc. These 'Add-On' Sales Items can have their own Sell Price specifically used when sold along with the originating Sales Line.

In this exercise we will try and promote **'Water Cooler'** whenever the Customer orders **'Spring Water'**.

Go to **Inventory>Items** and create Item **'WATER COOLER'** and enter a sell price of \$**150**. **'Save'** the record

Go to the **'List'** screen and select **'SPRING WATER'** then go to the Item **'Detail'** screen. 'Check' the **'Add-On Sales Apply'** checkbox. The adjacent **'Item Add-On Sales'** button will become active. Click on this button to bring up the **'Add-On Sale'** maintenance screen.

Click on **'Add'** to add a new 'Add-On Sale' Item and select **'WATER COOLER'** and enter the following:

Default Qty: Enter **1**

Special Pricing checkbox: 'check' this box

Special Price: enter special price of **\$125**

Sales Explanation: enter (say) **"The Water Cooler is currently on Special"**

'Save' the entry and exit the screen

Go into **Sales>Sales Orders** and select Sales Order **SO300013**. Go to the Lines screen and add a new line for **'SPRING WATER'**. You will see that - on the Info Line across the centre of the screen - a button (**Add-On Sale Items**) will appear. Click on this button.

If the Customer accepts the offer then simply click the **'Create Order Lines'** button. You will see that the **'WATER COOLER'** will be added to the Order at the special price.

Note: In the above example we used a stock 'Item' to offer as an Add-On Sale. This could easily be - for example - an extended Warranty, etc

2.9. Serial Numbered Items

When an Item is defined as Serial Number controlled Ostendo will ensure that each individual item has its own unique Serial Number. This Serial Number follows the Item through the full Stock Control and Sales Cycle process.

Go to **Inventory>Items** and click on the **'Add'** button. Create a new Item called (say) **'Camera'**. On the Item Detail panel 'check' the **'Serial No'** checkbox. Having defined this then:

- All receipts into stock for this Item must be given a unique Serial Number (Note: The same Serial Number can exist against a different Item Number)
- All Issues must be made from current stock. I.e. You are not allowed to issue a Serial Numbered Item that you don't have in stock!!!

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the Batch Line screen and add **'Camera'** with an **'Adjustment Type'** of **'Receipt'** and a quantity of **1**. If you try to **'Save'** the receipt then it will be rejected because a Serial Number is required. Enter a Serial Number in the line then click the **'Post All Adjustments'** button in the **'Detail'** tab.

If you now go into **Inventory>Inventory Availability** you will see that the individual Serial Number is displayed.

An Issue Transaction: To issue an item go into **Sales>Sales Orders** and select Order **SO300013**. Select the **'Lines'** tab and create a Sales Order Line for Item **'Camera'**. Click on the **'Picked Lines'** tab and then click the **'Add'** button. Place the cursor in the **'Qty'** field but do NOT enter a quantity. You should, however, click on the 'spyglass' icon to bring up a panel that shows the individual serial numbers in stock. Select the Serial Number then **'Save'** the Issue and **'OK'** to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Serial Number does not appear in Inventory. If you go to the **'Transaction History'** tab you will see that a Stock Movement record has been created to cover the Issue

2.10. Serial Kits

Serial Kits are individual – unique - Serial Numbers within an Item. Each Kit can comprise of one or more component Items that may vary in their individual attributes (Colour, Size, Batch, Serial Number, Expiry Date, etc). Ostendo will allow you to view each Serial Kit to determine what specific Items are included; plus the attributes those Items contain. This facilitates variable Kit contents within the same Serial Kit Code in addition to providing traceability of the individual Kit contents. An example of where this can be used is (say) a laptop computer. The computer itself is Serial Numbered and it contains a Hard -Disk Drive, DVD Burner, etc that have their own Serial Number in addition to a Warranty. Full visibility of the Serial Kit contents is visible at all times

This is how it works

2.10.1. Kit Lists

Kit Lists are templates that can optionally be used in association with Serial Kits. Whenever a Serial Kit Item is received into stock the Kit List is automatically attached to the Serial Kit Item record. These Lists are maintained via [Inventory>Lists](#). Go into this screen and create a Kit List called (say) '**Accessories**' add Items **6000-2200** (Colour Inkjet Printer) and **5000-5003** (LCD 21inch Monitor)

2.10.2. Serial Kits.

Go into [Inventory>Items](#) and create an Item called (say) '**Accessory Kit**'. While in that screen 'check' the '**Serial No**' checkbox. You should also 'check' the '**Serial Kit**' checkbox which identifies it as a Serial Kit Item. A field to the right of 'Serial Kit' will now become active from which you can optionally select the '**Accessories**' Kit List created above. This kit forms the base template that is attached to any receipt record.

2.10.3. Creating the Serial Kit

Whenever a specific Serial Number for the Item is received for the first time Ostendo will check to see if the Item is also flagged as a Serial Kit Item and, if so, a separate 'Serial Kit' record will be generated for the received Serial Number. To see this in action go into [Inventory>Inventory Adjustments](#) and create a new Adjustment Batch. In the 'Lines' tab add Item '**Accessory Kit**' with quantity of 1 and enter a Serial Number. Go back to the '**Detail**' tab and 'post' the adjustment.

2.10.4. Amending the Serial Kit content

If you now go into [Inventory>Serial Kits](#) you will see the generated Kit. You now have the option to amend the Kit Contents as required to:

- Identify specific variants such as Serial Number, Batch Number, etc
- Identify any specific additional user-defined information (E.g. Length)
- Add or delete Lines

In our example give the Printer and Monitor a specific Serial Number that is in the Kit. Note: This is not validated and is for information only

2.10.5. Serial Kit status

When the Serial Kit Item is sold the associated linked Serial Kit record will have its status automatically changed from '**In Stock**' to '**Issued**'. Full visibility of all the Serial Kits and their contents will be available at all times.

2.11. Expiry Date

When an Item is received into Inventory it can be given an Expiration or Use-By Date. Full visibility of Quantity by Expiry date is available. This Expiry Date follows the Item through the full Stock Control and Sales Cycle.

Go to **Inventory>Items** and select Item **1500-2187** (Blue Paint) and go to the **'Detail'** tab. On that screen 'check' the **'Expiry Date'** checkbox. Having defined this then:

- All receipts into stock for this Item must be given a Date
- All Issues from stock must contain an Expiry Date

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the **'Lines'** tab and add Item **1500-2187** with quantity of (say) **10**. If you try to **'Save'** the record then it will be rejected because an Expiry Date has not been entered. Enter an Expiry Date in the line. Add another line for the same Item with a different Expiry Date. Finally, go to the **'Detail'** tab and click on the **'Post All Adjustments'** button.

If you now go into **Inventory>Inventory Availability** and select Item **1500-2187** you will see the received Stock segregated by Expiry Date.

An Issue Transaction: To issue an 'Expiry Dated' item go into **Sales>Sales Orders** and select Order **SO300013**. Select the **'Lines'** tab and create a Sales Order Line for Item **1500-2187**. Click on the **'Picked Lines'** tab and then click the **'Add'** button. In the created line select the drop-down against the 'Qty' field to bring up a panel showing all stock for the Item. Select the specific Expiry Date and Quantity then **'Save'** the Issue and **'OK'** to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Quantity against the Expiry Date has been reduced by the issued quantity. If you go to the **'Transaction History'** tab you will see that a Stock Movement record has been created to cover the Issue

2.12. Revision Numbers

Revision control is the management of multiple revisions of the same unit. It is most commonly used in engineering to manage ongoing development versions of the same product. Changes are identified by incrementing an associated Revision Number (or Letter) code, termed the "revision number".

Go to **Inventory>Items** and select Item **FC-7620** (Filing Cabinet - 3 Drawer) and go to the **'Detail'** tab. On that screen 'check' the **'Revision No'** checkbox. An adjacent (Current Revision) field becomes active. Click on the right-most icon in this field to bring up **'Item Revisions'**. Add a couple of Revisions in that screen then **'Close'** it. Click on the Left icon in the **'Revision No'** field and select a Revision Level to represent the current Revision Level then **'Save'** the record

Having defined that this Item is subject to Revision Level Control then:

- All receipts into stock for this Item must be given a Revision Number
- All Issues from stock must contain a Revision

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the **'Lines'** tab and add Item **FC-7620** with quantity of (say) **5**. If you try to **'Save'** the record then it will be rejected because a Revision Level has not been entered. Go to the **'Revision Level'** field and select a Revision Level from the drop-down list. Add another line for the same Item with a different Revision Level. Finally, click the **'Post All Adjustments'** button in the Detail tab.

If you now go into **Inventory>Inventory Availability** and select Item **FC-7620** you will see that the received Stock is segregated by Revision Level.

An Issue Transaction: To issue an item go into **Sales>Sales Orders** and select Order **SO300013**. Select the **'Lines'** tab and create a Sales Order Line for Item **FC-7620**. Click on the **'Picked Lines'** tab and then click the **'Add'** button. In the created line select the drop-down against the **'Qty'** field to bring up a panel showing all stock for the Item. Select the specific Revision Level and Quantity then **'Save'** the Issue and **'OK'** to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Quantity against the Revision Level has been reduced by the issued quantity. If you go to the **'Transaction History'** tab you will see that a Stock Movement record has been created to cover the Issue

2.13. Batch Numbers

Batch Numbers enable the user to segregate stock by their Batch (or Lot) Number. The Food, Beverage, Chemical, and Pharmaceutical industries require this mandatory feature to control quality, quantity, and traceability. Batch/Lot Numbers can be combined with 'Expiry Dates' where required.

Go to **Inventory>Items** and select Item **1500-2186** (Red Paint) and go to the **'Detail'** tab. On that screen 'check' the **'Batch No'** checkbox. At this point you should note that Batch Numbers can be:

- Manually generated at time of receipt into Inventory, or
- Pre-keyed into a Batch List with option to also manually generate, or
- Pre-keyed into a Batch List from which the receipt Batch number MUST be selected

In the second two options you can click on the adjacent (**Batches**) button and pre-define the batches. You can restrict batch identities to those in this list by 'checking' the **'Restrict Batch Numbers to the List Below'** checkbox

Having defined that this Item is subject to Batch Control then:

- All receipts into stock for this Item must be given a Batch Number
- All Issues from stock must contain a Batch Number

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the **'Lines'** tab and add Item **1500-2186** with quantity of (say) **5**. If you try to **'Save'** the record then it will be rejected because a Batch Number has not been entered. Go along to the Batch Number field and either key in the Batch Number or select a Batch Number from the drop-down list. Add another line for the same Item with a different Batch Number. Finally, click the **'Post All Adjustments'** button in the **'Detail'** tab.

If you now go into **Inventory>Inventory Availability** you will see that the Stock is segregated by Batch Number.

An Issue Transaction: To issue an item go into **Sales>Sales Orders** and select Order **SO300013**. Select the **'Lines'** tab and create a Sales Order Line for Item **1500-2186**. Click on the **'Picked Lines'** tab and then click the **'Add'** button. In the created line select the drop-down against the **'Qty'** field to bring up a panel showing all stock for the Item. Select the specific Batch Number and Quantity then **'Save'** the Issue and **'OK'** to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Quantity against the Batch Number has been reduced by the issued quantity. If you go to the **'Transaction History'** tab you will see that a Stock Movement record has been created to cover the Issue

2.14. Colour Segregation

The Colour option allows a user to have many different colours against the same Item Code and

be able to segregate stock by their Colour. Within Ostendo this can be used independently or can be combined with Size and/or Grade to accommodate the Apparel industry who need Style/Colour/Size/Fit inventory segregation.

Go to **Inventory>Items** and select Item **485-2668** (Internal Spotlight 250 Watt) then go to the **Detail** tab. On that screen 'check' the **Colour** checkbox. At this point you should note that Colours can be:

- Manually generated at time of receipt into Inventory, or
- Pre-keyed into a Colour List with option to also manually generate, or
- Pre-keyed into a Colour List from which the receipt Colour number **MUST** be selected

In the second two options you can click on the adjacent (**Colours**) button and pre-define the available colours. You can restrict colour selections to those in this list by 'checking' the **Restrict Colours to the List Below** checkbox

Having defined that this Item is subject to Colour Control then:

- All receipts into stock for this Item must be given a Colour
- All Issues from stock must contain a Colour

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the **Lines** tab and add Item **485-2668** with quantity of (say) **5**. If you try to **Save** the record then it will be rejected because a Colour has not been entered. Go to the Colour field and either key in the Colour or select a Colour from the drop-down list. Add another line for the same Item with a different Colour. Finally, click the **Post All Adjustments** button in the **Detail** tab.

If you now go into **Inventory>Inventory Availability** you will see that the received Stock is segregated by Colour.

An Issue Transaction: To issue an item go into **Sales>Sales Orders** and select Order **SO300013**. Select the **Lines** tab and create a Sales Order Line for Item **485-2668**. Click on the **Picked Lines** tab and then click the **Add** button. In the created line select the drop-down against the **Qty** field to bring up a panel showing all stock for the Item. Select the specific Colour and Quantity then **Save** the Issue and **OK** to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Quantity against the Colour has been reduced by the issued quantity. If you go to the **Transaction History** tab you will see that a Stock Movement record has been created to cover the Issue

2.15. Grade Segregation

The Grade option allows a user to have many different Grades against the same Item Code and be able to segregate stock by their Grade (Example: Rough, Medium, Smooth)

Go to **Inventory>Items** and select Item **482-2218** (Standard 240 Volt Power Outlet - Single) then go to the **Detail** tab. On that screen 'check' the **Grade** checkbox. At this point you should note that Grades can be:

- Manually generated at time of receipt into Inventory, or
- Pre-keyed into a Grade List with option to also manually generate, or
- Pre-keyed into a Grade List from which the receipt Grade number **MUST** be selected

In the second two options you can click on the adjacent (**Grades**) button and pre-define the available Grades. You can restrict Grade selections to those in this list by 'checking' the **Restrict Grades to the List Below** checkbox

Having defined that this Item is subject to Grade Control then:

- All receipts into stock for this Item must be given a Grade
- All Issues from stock must contain a Grade

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the 'Lines' tab and add Item **482-2218** with quantity of (say) **5**. If you try to 'Save' the record then it will be rejected because a Grade has not been entered. Go to the Grade field and either key in the Grade or select a Grade from the drop-down list. Add another line for the same Item with a different Grade. Finally, click the 'Post All Adjustments' button in the Detail tab.

If you now go into **Inventory>Inventory Availability** for Item **482-2218** you will see that the received Stock is segregated by Grade.

An Issue Transaction: To issue an item go into **Sales>Sales Orders** and select Order **SO300013**. Select the 'Lines' tab and create a Sales Order Line for Item **482-2218**. Click on the 'Picked Lines' tab and then click the 'Add' button. In the created line select the drop-down against the 'Qty' field to bring up a panel showing all stock for the Item. Select the specific Grade and Quantity then 'Save' the Issue and 'OK' to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Quantity against the Grade has been reduced by the issued quantity. If you go to the 'Transaction History' tab you will see that a Stock Movement record has been created to cover the Issue

2.16. Size Segregation

The Size option allows a user to have many different Sizes against the same Item Code and be able to segregate stock by their Size. Within Ostendo this can be used independently or can be combined with Colour and/or Grade to accommodate the Apparel industry that need Style/Colour/Size/Fit inventory segregation.

Go to **Inventory>Items** and select Item **850-2192** (Clamp & Bearing) then go to the 'detail' screen. On that screen 'check' the 'Size' checkbox. At this point you should note that Sizes can be:

- Manually generated at time of receipt into Inventory, or
- Pre-keyed into a Size List with option to also manually generate, or
- Pre-keyed into a Size List from which the receipt Size number MUST be selected

In the second two options you can click on the adjacent (Sizes) button and pre-define the available Sizes. You can restrict Size selections to those in this list by 'checking' the 'Restrict Sizes to the List Below' checkbox

Having defined that this Item is subject to Size Control then:

- All receipts into stock for this Item must be given a Size
- All Issues from stock must contain a Size

A Receipt Transaction: One method of receiving stock is to go to **Inventory>Inventory Adjustments** and create a new Adjustment Batch. Go into the 'Lines' Tab and add Item **850-2192** with quantity of (say) **5**. If you try to 'Save' the record then it will be rejected because a Size has not been entered. Go to the Size field and either key in the Size or select a Size from the drop-down list. Add another line for the same Item with a different Size. Finally, click the 'Post All Adjustments' button in the 'Detail' tab.

If you now go into **Inventory>Inventory Availability** you will see that the received Stock is segregated by Size.

An Issue Transaction: To issue an item go into **Sales>Sales Orders** and select Order **SO300013**. Select the 'Lines' tab and create a Sales Order Line for Item **850-2192**.

Click on the **'Picked Lines'** tab and then click the **'Add'** button. In the created line select the drop-down against the **'Qty'** field to bring up a panel showing all stock for the Item. Select the specific Size and Quantity then **'Save'** the Issue and **'OK'** to exit the picking screen.

If you now go into **Inventory>Inventory Availability** you will see that the Quantity against the Size has been reduced by the issued quantity. If you go to the **'Transaction History'** tab you will see that a Stock Movement record has been created to cover the Issue

2.17. Combinations

You may wish to create your own Item code and give it a combination of the following characteristics.

- Serial Number
- Batch Number
- Colour
- Size
- Grade
- Expiry Date
- Revision Level

Carry out a Stock adjustment to receive some of the item into stock. Now add the Item to an existing Sales Order and issue the Item to the Order.

2.18. Item Additional Fields

There are two levels where Additional fields could be required against Items:

Global Fields that apply to ALL Items (Example:- SEATO Stock Number)

Properties that apply to some Items (Example:- Voltage)

2.18.1. Item Additional Fields

Go into **File>System Configuration>Additional Fields** and click on the **'Add'** button. On the displayed line enter the following:

Module: Select **'Items'**

Caption: Enter the Additional Field name (Example **'SEATO Stock No'**)

Field Type: From the drop-down list select the format of the field. The options are:

- **Text:** Any data format can be entered in a Text field
- **Decimal:** Allows entry of numbers and decimals
- **Integer:** Allows entry of whole numbers only
- **Currency:** Shows Currency symbol and decimals as defined in Regional Settings
- **Yes/No:** Shows a checkbox which can be checked/unchecked
- **Date:** Contains a drop-down calendar for selection of a date
- **Time:** Displays format HH:MM:SS for entry of a time of day

Value List: This allows you to define any specific entries to which a drop-down list - during data entry - is restricted

'Save' the entry and **'Close'** the screen when done

If you now go to the Item screen (**Inventory>Items**) and click on the **'Detail'** tab you will see a 'tab' (**Additional Fields**) in the centre-left of the screen. Click on this tab and enter some data into the **'SEATO Stock No'** field then **'Save'** the record.

You can view the Additional Fields in the Item List screen if required by going into the Item's List screen and 'right mouse' in the centre panel. Select **'Customize List Fields'** from the displayed panel. (Note: If that option is not visible then go to **File>System Configuration>User Security**

and Options and go to the 'User Options' tab for the current User. 'Check' both the 'Save Grid Layouts' and 'List Customising' checkboxes.)

On the displayed panel 'check' **Additional Field1** and give it a 'Display Name' of '**SEATO Stock Number**'. 'Save' the entry. The field will now display on the List screen where you can sort and filter as necessary.

2.18.2. Item Properties

This feature allows you to define a 'Property' (Example: Voltage) and then link that property to selected Items with a value that is specific to each Item (Example 230 Volts)

To demonstrate this go into **General>Settings>General Properties** and add '**Voltage**' with 'Property Type' of '**Text**' and the following entries - on separate lines - in the Property Values field (**115 Volts**, and **230 Volts**). Click on '**Save**' and then '**Close**'

Now go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the '**Related**' Button on the right of the screen and select '**Item Properties**'. Click the '**Add**' button and:

- Add a line using the above property
- Select the Voltage from the drop-down list
- Select the specific Voltage from the drop-down under column '**Value**'
- 'Check' the '**Copy to SO Lines**' checkbox

The selected property and value will accompany the Item whenever it is used in a Sales Order. To demonstrate this go into **Sales>Sales Orders** and select Order **SO300013**. Select the '**Lines**' tab and create a Sales Order Line for Item **485-2267**. Click on the '**Line Properties**' tab and the Properties will have been copied from the Item to this Sales order Line

2.19. Item Images

You can add multiple images (pictures, drawings, maps, plans, etc) to an Item. These can be printed on all documents where the Item is used. Go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the '**Related**' Button on the right of the screen and select '**Item Images**'. Click the '**Add/Edit**' button and:

- Give the Image a short Name
- Point the program to where the image is located on your computer network
- 'Check' the '**Copy to Assembly Order**' box then '**Save**' and '**Exit**' the screen
- Go into **Assembly>Assembly Order** and select an Order whose status is '**Open**' or '**In Progress**'. Click on the '**Lines**' tab and add Item **485-2267** to the Order
- If you now click on the '**Related**' button and select '**Job Line Images**' you will see that the Image linked to the Item has now been copied to this Assembly Order Line.
- Go into the '**Detail**' tab and click on the '**Assembly Sheet**' Icon to print the Assembly sheet.

2.20. Item Documents

You can add multiple documents to an Item. These can be printed along with all documents where the Item is used. Go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the '**Related**' Button on the right of the screen and select '**Item Documents**'. Click the '**Add/Edit**' button and:

- Give the Document a short Name
- Point the program to where the document is located on your computer network
- 'Check' the '**Copy to Assembly Order**' box then '**Save**' and '**Exit**' the screen
- Go into **Assembly>Assembly Order** and select an Order whose status is '**Open**' or '**In Progress**'. Click on the '**Lines**' tab and add Item **485-2267** to the Order

- If you now click on the '**Related**' button and select '**Job Line Documents**' you will see that the document linked to the Item has now been copied to this Assembly Order Line

2.21. Item Customers

Your Customers may order Items that refer to the same Item you hold in stock - but know it by a different Item Code. You can create these cross-references in Ostendo. This provides an added benefit in that you can enquire on your Items and that enquiry will include all Customer Identities. To create a Customer cross-reference go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the '**Related**' Button on the right of the screen and select '**Item Customers**'. Click the '**Add/Edit**' button and:

- Select the Customer from the drop-down list
- Enter the Customer's equivalent Item Code and Description
- Enter the Unit of Measure by which the Customer normally orders the Item
- Enter a conversion factor that converts from the Customer's Unit to the Item's Base Unit. Example: If Base Unit is EACH and the Customer's Unit is TEN then the conversion factor would be 10

If you now go to **File>System Configuration>System Settings** and 'check' the '**Advanced Searching**' checkbox then whenever you are in a '**List**' view of Items and you enter the Customer's Item Code into the '**Search**' Box at the bottom of the screen it will find your Ostendo equivalent Item Code.

2.22. History Notes

This function allows you to link multiple time-stamped notes to an Item. Against selected History Notes you can also add a dated reminder so that Ostendo will prompt you of the reminder once the date is reached. Go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the '**Related**' Button on the right of the screen and select '**Item History Notes**'. Click the '**Add**' button and:

- Enter some history notes
- 'check' the '**Follow-Up**' required checkbox and select a date from the adjacent drop-down calendar. '**Save**' the History Note and exit the screen
- To see the '**Follow Up**' in action you should first change the company by clicking on **File>Change Company** and selecting **DEMO** then sign in as **ADMIN/pass**. Repeat this and go back to company '**Training**'. Upon sign-in as **ADMIN/pass** the alert should present itself
- To print the History Notes go into the Item screen and highlight the Item for which you want to print the History Notes. Click on the '**Reports**' button on the right of the screen and select '**Item History Notes**'

3.2 Descriptors

A Descriptor is simply a User-defined entry that represents anything that you wish to Invoice. It contains a Sell Price, Buy Price (if Purchased), Cost, and a Tax Group Code to define the Tax to be charged. Examples of Descriptors are:

- Non-Stock Items
 - Travel
 - Accommodation
 - One-off charges
 - Labour Charge
 - Sub Contract charge
 - etc
-

1. Setup

The Descriptor record contains fields that require completion by selecting from a user-defined list. You should have a look at these fields before creating your first Descriptor record.

Units of Measure: Each Descriptor record MUST be given a Unit of Measure. A short list of popular Units are in the base Ostendo. Add your own Units of Measure by going into **General>Settings>Standard Units**

Tax Group: Each Descriptor record MUST be given a Tax Group. This allows the combination of Descriptor's Tax Group and Customer's (or Supplier's) Tax Group to denote the Tax Code (and therefore the Tax Rate). You can maintain Tax Groups via **File>Financial Configuration>Tax Groups**. Within the Tax Group screen you can nominate a 'Default' Tax Group that will populate a Descriptor record during creation. (You can, of course, amend the Tax Group in the created record). It is suggested that, for now, use the default Tax Group when creating a new Descriptor record.

Categories: This is used extensively for selection criteria when updating or Analysing data or printing reports. This is not essential and can be done at any time. To create and maintain Categories go into **Inventory>Settings>Categories**. It is not essential to look at Categories at this stage, as they will be covered in other Training Sessions where they will be utilised.

Pricing Groups: Pricing Groups are used to facilitate 'Batch Update' of Sell prices using screen **Pricing>Batch Price Update**. If you will be using the Batch Price Update facility then create the Pricing Groups via **Pricing>Settings>Pricing Groups**. It is not essential to look at Pricing Groups at this stage, as they will be covered under the **Sell Price, Buy Price and Costs** Training session

Analysis Groups: Analysis Groups are used across Ostendo to facilitate viewing and reporting by Group. Additionally the Groups are used to format the print content within Quotes and Invoices. If you are going to use these features then you should first establish the Analysis Groups by going to **General>Analysis Groups**. It is not essential to look at Analysis Groups at this stage, as they will be covered in other Training Sessions where they will be utilised.

2. Create a Descriptor

Go to **Inventory>Descriptors** and click on the 'Add' button. Create a new Descriptor called (say) '**Sub Contract**'. Select UOM of '**Hrs**' from the drop-down list against field '**Unit**'. Enter a description (Example: '**Sub Contract Labour**'). Enter a '**Std Sell Price**' of (say) **\$50.00**. Click the '**Save**' button to save the Descriptor to file.

To see how Descriptors are used go into **Sales>Sales Orders** and select Order **SO300013** and go to the '**Lines**' tab. Click on the 'Add' button and, in the lower part of the screen select '**Descriptor Code**' from the drop-down against '**Line Type**'. Now select the above Descriptor from the drop-down list against '**Code**'. Use an '**Order Qty**' of (say) **5** hrs then click the '**Save**' button.

Go to the Order '**Detail**' tab and then click on the '**Printer**' Icon to the right of the '**Acknowledgement Status**' field and view the Order Acknowledgement. The whole Sales Order process steps will be addressed within the **Customers, Sales Orders and Invoicing** Training document

Let's now look at other fields that are held against a Descriptor record. Therefore, go back to

Inventory>Descriptors and select the Descriptor that you created above.

2.1. Status

Ostendo has 2 statuses used by Descriptors

Active - Can be used throughout Ostendo

Inactive - Cannot be seen or selected in any drop-down lists of Descriptors

Change the **Status** to **Inactive** then go back to **Sales>Sales Orders**. Select Sales Order **SO300013** and click on the **Lines** tab. If you try and add this Descriptor to the Sales Order line you will see that it is not visible in the drop-down list

Go back into **Inventory>Descriptors** and you will see that - by default - Inactive Descriptors are not visible. You can include them in the list by 'checking' the **Include In-Active Status** checkbox at the bottom of the screen. Select the Descriptor that you created above and change the status back to **Active**

2.2. Source By

This field defines from where the Descriptor is sourced: The displayed options are:

Purchasing - This means that the Descriptor will automatically be available to Purchasing whenever a requirement arises.

Internal - This means that the Descriptor will automatically place a demand on In-house resources whenever a requirement arises

The use of these options will be explained in more detail in the **Inventory Control** Training document. For now leave the **Source By** set to **Purchasing**.

Whenever the **Source By** is set to **Purchasing** it is strongly recommended (but not essential) that you also select a default Supplier from the drop-down list of Suppliers. Therefore click on the drop-down against field **Primary Supplier** and select a Supplier (say) **Bruce Wilson**.

2.3. 'On-the-fly' Purchased Descriptor added to Orders

You can receive a Descriptor from a Purchase Order and immediately issue it directly to a Job Order or Assembly Order even though it was not planned for the Order. To see this in action:

Go to **Inventory>Descriptors** and click on the **Add** button. Create a new Descriptor called (say) **General**. Select UOM of **Each** from the drop-down list against field **Unit**. Enter a description of (say) **General Description** (This will be overwritten during the Purchase Receipt). 'Check' the **Freeform: Receipts create new Job or Assembly Lines** checkbox then click the **Save** button to save the Descriptor to file.

Go to **Purchasing>Receipts** and click on the **Add** button. On the presented screen:

Select **Receipts No Order** from the drop-down under **Receipt Style**

Select any Supplier from the drop-down under **Supplier**

Click the **Save** button

Click on the **Lines** tab to go to the Receipt Lines

In the Receipt Lines screen enter the following

Click the **Add** button and then refer to the lower part of the screen

Select **Descriptor Code** from the drop-down under **Line Type** field

Select **General** from the drop-down under **Code** field

Enter a **Receipted Qty** of (say) **1**

Enter a **Receipt Unit Price** of (say) **\$100**

Amend the **Description** to (say) **Consultancy Work as directed**

The next step is to link the receipt to where it is to be used. Therefore

Click on the '**Allocations**' tab

Select '**Job Order**' from the drop-down under '**Allocation Type**'

Select any Job from the drop-down under '**Allocation Reference**'

Click on the '**Yes**' button on the presented panel to create the Job Line

Click the '**Detail**' tab at the top of the screen and click the '**Post all Purchase Receipt entries**' button

Finally let's have a look at what has happened within the Job

Go to **Jobs>Job Orders** and select the Job to which the above Receipt was posted

Click on the '**Lines**' tab and you will see that the Descriptor (with the amended Description and Received Qty/Price) has been added to the Order

2.4. 'Cost Plus' Descriptors for Job Orders

In addition to receiving the Descriptor as described in 2.2.4. you can also (for Job Orders) specify if you use the actual cost of the receipt to calculate the Sell Price against the Job Order Line base on a pre-defined Markup. To see this in action:

Go to **Inventory>Descriptors** and select '**General**' then go to the '**Details**' tab. 'Check' the '**Calculate Sell Price from Receipt Cost for Jobs**' checkbox and enter (say) **100** in the adjacent '**Markup**' field. Click the '**Save**' button to save the changes.

Repeat the process flow that you carried out in 2.2.4.

You will that, in the Job Line, the '**Unit Price**' of the Descriptor is the Descriptor's Purchase Price with 100% markup

2.5. Sales Warranty

You can attach a Warranty Identity to a Descriptor. That Warranty is carried through to Sales and Job Orders and becomes effective when the Descriptor's usage is recorded against the Order

Go into **Service>Warranty Definitions** and view the Warranty Codes that are currently on file. Add your own as required.

Go back to **Inventory>Descriptors** for Descriptor '**Sub Contract**' and click on the '**Sales Warranty Applies**' checkbox. From the drop-down list in the adjacent field select the above Warranty. '**Save**' the Descriptor record.

Go into **Sales>Sales Orders** and select Sales Order **SO300015** and click on the '**Lines**' tab. Click on the '**Add**' button and, in the lower part of the screen add Descriptor Code '**Sub Contract**' with quantity (say) **4** and '**Save**' the entry. If you now click on the '**Warranty**' tab in the lower part of the screen you will see that the Warranty Code has been copied from the Descriptor record to the Sales Order Line.

Whilst you are focused on this Sales Order Line click on the '**Picked Lines**' tab, click on the '**Add**' button and issue the Descriptor. '**Save**' the record then '**Close**' the Sales Order screen then go into **Service>Warranty List** where you will see the generated Warranty record. You can, of course, amend the details as required.

3.3 Labour Codes and Employees

You can define Labour at two levels of complexity dependant upon your Operational requirements. These are:

If you simply want to include Labour against which you record actual hours used in Orders then create '**Labour**' as a Descriptor Code (described in the previous section) and ignore this section.

If, on the other hand, you are going to:

- Plan and/or schedule Labour activities within a Job
- Plan and/or schedule Employee activities within a Job
- Book Employee times to specific Labour Activities
- Record Employee Timesheets

Then you should create and maintain **Labour Codes** and **Employees** as described in this section

1. Setup

The **Labour Code** record contains fields that are completed by making a selection from a user-defined list. You should have a look at these fields before creating your first **Labour Code** record.

Tax Group: Each Labour Code record MUST be given a Tax Group. This allows the combination of Labour Code's Tax Group and Customer's (or Supplier's) Tax Group to denote the Tax Code (and therefore the Tax Rate). You can maintain Tax Groups via **File>Financial Configuration>Tax Groups**. Within the Tax Group screen you can nominate a 'Default' Tax Group that will populate a Labour Code record during creation. (You can, of course, amend the Tax Group in the created record). It is suggested that, for now, use the default Tax Group when creating a new Labour Code record.

Categories: This is used extensively for selection criteria when updating or Analysing data, or printing reports. This is not essential and can be done at any time. To create and maintain Categories go into **Inventory>Settings>Categories**. It is not essential to look at Categories at this stage, as they will be covered in other Training Sessions where they will be utilised.

Departments: Departments are used to group Labour Tasks. If you are going to analyse Labour loading by Department then you should first create the Departments via **General>Settings>Departments**. It is not essential to look at Departments at this stage, as they will be covered in the **Job Orders** Training Sessions where they will be utilised.

Analysis Groups: Analysis Groups are used across Ostendo to facilitate viewing and reporting by Group. Additionally the Groups are used to format the print content within Quotes and Invoices. If you are going to use these features then you should first establish the Analysis Groups by going to **General>Analysis Groups**. It is not essential to look at Analysis Groups at this stage, as they will be covered in other Training Sessions where they will be utilised.

2. Labour Codes

Labour Codes are used in Quotes, Assembly Orders and Jobs to determine the **Planned** Sell Price and Cost of any 'Labour' activity. An Employee subsequently books time against a Labour Code to provide **Actual** Labour Costs for the Activity

To set up **Labour Codes** go into **Labour>Labour Codes** and create (say) '**Electrician**' with a Unit

of Measure of '**Hrs**'. Also enter a nominal '**Standard Rate**' that denotes the normal Charge Rate and '**Standard Cost**' which indicates the nominal Cost of using this Labour Code. '**Save**' the record

Note: The Standard Cost can optionally be broken down into:

- Direct Labour Cost
- Variable Overhead (Expressed as a percentage of the direct labour cost)
- Fixed Overheads (Expressed as a monetary value per hour)

3. Employees

To set up Employees go into **Labour>Employees** and create yourself as an **Employee**. The significant field to note are:

- **Email Address**: Enter your email address. This will be used in later exercises when emailing documents
- **Employee Unit Cost**: Enter the hourly Cost of your services of (say) **\$20**. This will be addressed further in the Employee to Labour Code section (below).
- **Sales Person and Buyer**: 'Check' the checkboxes to identify yourself as both a **Salesperson** and a **Buyer**. This means that you will appear in the drop-down list against Salesperson in the main Customer screen and as a Buyer in the main Supplier screen.

4. Employee to Labour Code

This feature allows you to define different costs whenever an **Employee** books against different Planned **Labour Code** activities. To set up the relationships go into **Labour>Employees** and highlight the **Employee** record that you have just created. Click on the '**Labour Codes**' tab then click on the '**Add**' button.

Select '**Electrician**' from the drop-down list under '**Labour Code**'

Enter a **Fixed Overhead** Value of (say) **\$10**

Enter a **Variable Overhead** Percentage of (say) **25%**

You will notice that the field '**Total Cost Per Unit**' is now populated with the system-calculated value taken from the entries made. It is this cost that is applied when you (as the **Employee**) books time against an activity Planned for this **Labour Code**

Create a second link to another **Labour Code**.

Now go back to the '**Detail**' tab and in the field '**Default Labour Code for Charging**' select '**Electrician**'. This denotes the **Labour Code** that will be used if you book time against a Job or Assembly Order where the activity was not previously planned.

Now let's see how all this comes together.

5. Process Flow for Labour/Employees

Having created an Employee and established a relationship to a Labour Code the following is how this is applied when used in a Job or Assembly Orders

Go into **Jobs>Job Orders** and select Job **JOB400002** and then click on the '**Lines**' tab for that order. Click the '**Add**' button and, in the lower part of the screen, select '**Line Type**' **Labour Code** and then select the above Labour Code in the drop-down against '**Code**'. Enter a quantity of (say) **5** and '**Save**' the record. You will notice that the '**Unit Price**' is the Standard Rate as entered against the Labour Code and the '**Planned Unit Cost**' is the Standard Cost as entered against the Labour Code.

Now go to the '**Actual Issues**' tab against the Order Line and enter **5** hours of actual time and select the Employee that you used above. If you click the '**Save**' button you will notice that the line's **Unit Cost** will adjust to the cost of the **Employee/Labour Code** relationship established above.

6. Timesheets

Go to **Labour>Timesheets**. You may also enter employee times via this screen. Click on the '**Add**' button and select the Employee created above then select the '**Lines**' tab. After confirming the Batch creation you will be presented with the Timesheet Line entry screen. Click the '**Add**' button and enter a booking as follows:

Type: **Job**
Reference: **JOB400002**
Hours: (say) **4**
Task or Step: **WindowsTraining**
Labour Code: **Electrician**

'**Save**' the entry and go back to the '**Detail**' view of Timesheet. In that view click the '**Post all Timesheet entries**' to post the batch.

Now go back to **Jobs>Job Orders** and go to the Lines tab of Job **JOB400002**. You will see that because **Labour Code Electrician** was not planned against Task **WindowsTraining** then a line has been added with a **Planned Order Qty** of **0** but an **Actual Qty** of **4**

4 3. Inventory Control

Ostendo has a comprehensive Inventory Control system that includes:

- Multi-Warehouse - Multi-Location
- Visibility of stock by multiple Units of Measure
- Visibility of Stock by Item variants (Serial No, Lot No, Colour, Expiry Date, etc)
- Stock Movement History maintained
- Allows negative stock
- Interactive Stock Replenishment routine
- Suggested Replenishment Order conversion to Purchase and/or Assembly Orders
- Miscellaneous Stock Adjustments
- Inventory Transfer
- Location Restock
- Inventory Stock Count
- etc

4.1 Warehouses and Locations

1. Warehouses

Ostendo has the ability to maintain multiple Warehouses. These Warehouses can be 'In House', at a Customer location, or at a Supplier location. The Warehouse can be flagged to define if its stock can be excluded from the Stock Replenishment routine. To create a Warehouse go into [Inventory>Warehouses](#) and click the 'Add' button. Enter the following:

Warehouse Code: enter your new Warehouse Code

Description: enter a short description of the Warehouse

Company: Select 'Site'

Name: Select 'Company'

You will notice that you can select a Customer or Supplier as the 'Company' in which case the 'Name' drop-down will show the Customers or Suppliers on file. By defining a Customer or Supplier Warehouse you can control your Consignment Stock at those locations

2. Locations

Within each Warehouse you can maintain multiple locations. An inventory Item can exist in multiple locations across multiple Warehouses. A location can be included in a user-defined Group and Sequence that facilitates Inventory checking in a logical, user-defined, sequence. To add a Location to a Warehouse go to [Inventory>Locations](#) and click the 'Add' button. Create a Location in your Warehouse.

Location: enter (say) **BIN01**

Description: enter a short description of the Location

Warehouse: From the drop-down list select the Warehouse you created above

3. Default Locations

Each Item can be given a Default Warehouse/Location for both Issues and Receipts. This is used to prefill the fields when receiving and issuing them and also for auto-issuing when contained in a Kitset. It can also be used as a global default for those users who do not use Warehouses or Locations.

If you go into [Inventory>Settings>Item Rules](#) you can specify a Global Default Warehouse and Location. Go into [Inventory>Items](#) and click the 'Add' button. Create a new Item called (say)

BOLT and click the 'Create' button. On the displayed '**Items Detail**' screen enter data into the following fields (i.e. They are a mandatory requirement against Item records)

Unit: Select a Unit of Measure from the drop-down list

Description: Enter a short description of the Item

'**Save**' the record. If you now click on the '**Additional Inventory Settings**' button you will see that this default Warehouse/Location is included in the record. Of course you can amend it for this Item if you wish.

4.2 Inventory Movements

Ostendo covers the following aspects of Inventory:

- Allows negative stock where applicable
- Simple Issue and Receipt functions throughout the system
- Miscellaneous stock movements.
- Inter-Warehouse transfers
- Re-stock processing by Warehouse
- Stock Replenishment routines
- Cyclic Stock Checks
- View current Stock by Warehouse/Location/Item/Item Variable
- View Stock Movement History
- View Stock Valuation (Standard/Average/Last)

The following Stock Movement activities are included in Ostendo:

- 'Formal' Stock Movements are covered in the following areas
 - Assembly Order Component Issue
 - Assembly Order Product receipt
 - Sales Order Line Issue
 - Job Order Line Issue
 - Purchase Order receipt

These stock movements will be covered under the Assembly, Sales, Jobs and Purchase Order Training sessions

- Other Stock Movements covered are:
 - Miscellaneous Stock Adjustments
 - Inventory Transfers
 - Inventory Restock
 - Stock Counts

These movements will be addressed below

1. Negative Stock Settings

'Out of the box' Ostendo allows any Item except Serial Numbered Items to go into negative Stock.

- If you do not want to allow negative stock in your business then you can set this system-wide by going into **Inventory>Settings>Item Rules** and 'check' '**Disable Negative Stock for All Items**' checkbox.
- If, on the other hand, you have specific Items that are not allowed to go negative then go into **Inventory>Settings>Item Rules** and 'uncheck' '**Disable Negative Stock for All Items**' checkbox. Now go into the Items screen (**Inventory>Items**) and select the specific Item that is not allowed to go negative; click on the '**Additional Inventory Settings**' button. In the presented panel 'check' the '**Prevent Negative Stock for this Item**' checkbox.

2. Miscellaneous Stock Adjustments

This option should be used where Inventory is being issued to, or received from, sources outside of Ostendo's formal processes. It can also be used to make instantaneous adjustments where the 'location' stock is found to be at variance to the 'Book' stock.

Go into *Inventory>Inventory Adjustments* and click the 'Add' button to create a new batch. 'Save' the batch then click on the 'Lines' tab and enter the following:

- Click on the 'Add' button to create a new line record
- Click on the drop-down against 'Item Code' and select the Item that you created above. Note: This will bring forward the Item's default Warehouse/Location but this can be amended if required.
- Select Adjustment Type 'RECEIPT'
- Enter an Adjustment Qty +/- of 12 (Note: Enter a positive amount for receipt, and a negative amount for Issue)
- Click the 'Save' button
- Repeat for other adjustments if required
- Go back to the 'Details' tab and click the 'Post all Adjustments' button

If you now go to the Item record in *Inventory>Items* you will see that 'On-Hand Qty' has been increased to reflect the adjustment. If you click on the 'Inventory Availability' button to the right of the 'On-Hand Qty' then a screen will appear that shows:

- Quantity held by Location
- Click on 'Transaction History' tab to view stock movements
- Click on 'Projected Availability' tab to view projected stock balances taking into account all supply and demand orders in Ostendo
- Click on 'Order Details' tab to view the current Supply and Demand orders

3. Inventory Transfer

The Inventory Transfer function covers the Batch transfer of stock from one location to another. Each line allows you to specify both the 'From' and 'To' Warehouse/Location with a quantity being transferred.

Go into *Inventory>Inventory Transfer* and 'Add' a new batch. After saving the batch click on the 'Lines' tab and enter the following:

- Click on the 'Add' button to create a new line record
- Click on the drop-down against 'Item Code' and select the Item that you received above.
- Enter all relevant information such as
 - From Warehouse/Location
 - To Warehouse Location
 - Quantity and Unit of measure
- Click the 'Save' button
- Repeat for additional lines if required
- Go back to the 'Details' tab and click the 'Post all Transfers' button

If you now go to the Item Availability screen *Inventory>Inventory Availability*:

- You will see the Quantity held by Location
- Click on 'Transaction History' tab to view stock movements. You will note that two transactions cover the transfer. I.e. Issue from the 'old' location and Receipt into the 'new' location

4. Inventory Restock

The Inventory Restock function covers the replenishment of stock relative to an Item's Re-Order Level and Replenishment quantity at specific Warehouses. This creates a Transfer List of affected Items so that stock can be replenished from a central Warehouse.

In this example we will create a restocking Template to apply to a delivery van and replenish stock into that van

Step 1. Create a Restocking Template by going into *Inventory>Settings>Restocking Templates* and click the 'Add' button. Give the Template a name (say) 'Van Restock'. Enter a description, and then click on the 'Levels' tab. Add the following Lines

<u>Item</u>	<u>Unit</u>	<u>Re-Order Level</u>	<u>Transfer quantity</u>
100-2002	Each	100	300
100-2008	Each	100	250

Step 2. Create a Warehouse (say) **Van1234** as described in 1.
Create a Location in that Warehouse (say) **Van Stock** as described in 2.

Step 3. Go to *Inventory>Location Restock* and click the 'Add' button. Enter the following:

- **From Warehouse:** Main
- **From Location:** Primary
- **To Warehouse:** Your Van Warehouse
- **To Location:** Van's Warehouse Location
- Check the 'Use Inventory Template' checkbox and select the Template created above
- Click the 'Save' button
- At this point the program will evaluate the current stock in the Van location and, if it is below the 'Re-Order Level' it will generate a record for the 'Re-Order Quantity'
- Click on the 'Lines' tab to view the calculated quantities. You can amend these if required.

Step 4. Click on the 'Reports' button and print the 'Inventory ReStock Sheet'. This will be used as a Pick List to pick the Items from stock

Step 5. Click on the 'Transfer all Items where Actual Transfer Qty <> 0' button. This will immediately 'post' the transferred quantities to the Van Location.

If you now go to the Item Availability screen *Inventory>Inventory Availability*

- You will see the Quantity held by Location
- Click on 'Transaction History' tab to view stock movements. You will note that two transactions cover the Restock. I.e. Issue from the 'old' location and Receipt to the 'new' location

5. Stock Counts

You can generate your own stock count from the following parameters:-

- From/To Warehouse
- From/To Location
- From/To Category
- From/To Item
- Cycle Count Code
- ABC Class

Multiple Cycle counts can be open at any time

To carry out a Stock Count do the following:

Step 1. Go to *Inventory>Inventory Count* and click the 'Add' button

On the displayed panel enter a Count Description then click the 'Generate Inventory Count Lines' button. You will note that for each Item Variant (Colour, Size, Serial Number, etc) a separate record is created for you to check the current stock.

Step 2. Click on the 'Reports' button and print the 'Inventory Count Sheet'. This sheet is then used to check the current Stock levels

Step 3. Enter the Stock Levels into field 'Count Qty'. Note: If you find other Items in stock or the selected Items in other locations then you can add the record by clicking the 'Add' button and entering the details.

Step 4. After re-printing and validating the entries you should go to the 'Detail' tab and click on the 'Update Counted Inventory Lines' button

If you now go to the Item Availability screen *Inventory>Inventory Availability* for one of the above Items:

- You will see the new Quantity in the specified Location
- Click on 'Transaction History' tab to view stock movements. You will note that a single Count Transaction has been generated to cover the Count adjustment quantity

4.3 Inventory Replenishment

Two options are available to Replenish Inventory:

Single Level MRP: This focuses on nominated Items and creates Suggested Orders required to maintain stock to satisfy the future actual and/or Forecast demand.

Multi-Level MRP: In addition to looking at the current actual and/or forecast future demands this option also looks to see if the Item is a Parent in a Bill of Material and, if so, will 'explode' any Suggested Orders to determine the components required to make it. These are then used as 'Suggested Demands' when assessing component requirements. To see how each of these work you should prepare the data as follows

1. Single Level MRP

1.1. Preparation

Go into *Inventory>Items* and select Item **100-2000** (Washer-Mild Steel-8mm). In that record you will see 4 fields that are used in the Replenishment function

Re-Order Level: If the projected stock falls below this level then a Suggested Order is triggered using the Re-Order Quantity or the actual demand quantity (whichever is the greater) to bring the stock up to the Re-Order level

Re-Order Qty: The minimum quantity that will be generated by a Suggested Order

Order Multiple: The Suggested Order quantity is rounded up to the nearest multiple of this entry

Lead Time: The number of days between when the Item should be ordered and when it is required

Amend the current entries against Item **100-2000** if required

Item Forecast: A Forecast can optionally be held against an Item and allows you to create a 12-month Forecast. The forecast contains individual monthly forecasts where each

month can selectively contain forecasts based on

- A Daily Demand for a 5-Day Week
- A Daily Demand for a 6-Day Week
- A Daily Demand for a 7-Day Week
- A single Monthly Demand

To create a Forecast go into **Inventory>Item Forecast**

1.2. Replenishment Process

Step 1. To commence the Replenishment process go to **Requirements>Inventory Replenishment**. In the panel that is presented:

- Click on the **'Create New Inventory Replenishment'** button
- 'Check' the **'Exclude any Forecasts for Items'** checkbox ***
- Select **'From Item Code' = 100-2000**
- Select **'To Item Code' = 100-2000**
- Click the **'OK'** button

*** To understand the basic Replenishment routine it would be better to ignore any Forecast at present. Having understood the process without a Forecast you may wish to repeat the Inventory Replenishment including the Item's Forecast.

Step 2. Determine Nett Demand

The Replenishment Routine extracts all 'Actual Demands' from the following sources

- Sales Orders
- Assembly Orders
- Jobs

The Forecast demands and the Actual Demands are then compared such an ongoing demand is generated from the greater of the Cumulative Forecast and the Cumulative Actual Demand. For Example:

Forecast	10	10	10	10	10	10
Actual	15	3	14	2		

Cumulative	15	5	12	8	10	10

The above analysis is carried out up to a Replenishment Horizon. This was defined in the run selection parameters in Step 1. The parameter can be either:

- A fixed number of Days for all Items
- An Item's Leadtime + A defined number of days

Step 3. Determine Supply

The Replenishment Routine extracts all 'Actual' Supply quantities from the following sources

- Purchase Orders
- Assembly Orders

Step 4. Creating Suggested Orders

Armed with the current stock level, Nett Demands, and Outstanding Supply orders then - starting the system date - the Replenishment program will go through the following process:

1. Add any overdue Supply Orders to the Current Stock Level and deduct any Nett Demands equal or prior to the system day

2. Is the 'expected' Stock Level below the Item's Re-Order Level?
 - No** - Go to question 4
 - Yes** - Determine the greater of (a) the quantity required to bring it up to the Re-Order Level and (b) the Re-Order Quantity. Round this up in accordance with the Item's Order Multiple and create a Suggested Order
3. Add the Suggested Order Quantity to the Current Stock Level
4. Go to the next day
5. Has the Replenishment Horizon been reached for this Item?
 - No** - Go to question 6
 - Yes** - End the Replenishment calculation for this Item
6. Add any Supply Orders (Assembly Orders or Purchase orders) scheduled for this day
7. Deduct any 'Nett Demands' scheduled for this day.
8. Go back to the question 2

Step 5. Replenishment Results

The results of the Replenishment run are displayed showing all Items that have had Suggested Orders created. You can drill-down on each Item (clicking the 'Detail' tab) to see the detailed results of the run. This shows (by Day) all Demands, Supply Orders, and Suggested Orders with a resulting stock balance at the end of the day. On this 'detail' view the quantity displayed against the Suggested Order can be amended. The results of the change are immediately reflected on all subsequent daily balances.

If you have made all the required changes then go back to the '**List**' view, 'check' the lines that are to be actioned and click the '**Accept Replenishment results for selected items**' button. The created 'Suggested' Order are converted into physical orders via **Inventory>Create Required Orders**

2. Multi-Level MRP

Let us look at a simple example to demonstrate Multi-Level MRP. Go into **Inventory>Items** and create two Items with data shown below

Item Code	A
Unit	Each
Description	Top Assembly
Supply Method	Supply from Stock
Sourced By	Assembly
Reorder Level	1
Reorder Qty	4
Order Multiple	1
Leadtime	7
Item Code	B
Unit	Each
Description	Purchased Component
Supply Method	Supply from Stock
Sourced By	Purchasing
Reorder Level	1
Reorder Qty	10
Order Multiple	1
Leadtime	14

The next step is to create a Bill Of Material. Go into **Assembly>Bills Of Material** and create a new BOM for Item **A**. On the presented BOM screen click on the '**Lines**' tab and add a component **B** with a usage quantity of **2**

Now run multi-level MRP as follows. Go into **Requirements>Inventory Replenishment** and, in the parameters screen, 'check' the Multi-level explosion checkbox.

Click the '**OK**' button to commence the Multi-Level Replenishment run

The program will go through the following steps;

Step 1. Determine Nett Demand

The Replenishment Routine extracts all 'Actual Demands' from the following sources

- Sales Orders
- Assembly Orders
- Jobs
- Suggested Assembly Order Components *** (See later)

The Forecast demands and the Actual Demands are then compared such an ongoing demand is generated from the greater of the Cumulative Forecast and the Cumulative Actual Demand. For Example:

Forecast	10	10	10	10	10	10
Actual	15	3	14	2		
Cumulative	15	5	12	8	10	10

The above analysis is carried out up to a Replenishment Horizon. This was defined in the run selection parameters in Step 1. The parameter can be either:

- A fixed number of Days for all Items
- An Item's Leadtime + A defined number of days

Step 2. Determine Supply

The Multi-Level MRP Routine extracts all 'Actual' Supply quantities from the following sources

- Purchase Orders
- Assembly Orders

Step 3. Creating Suggested Orders

Armed with the current stock level, Nett Demands, and Outstanding Supply orders then - starting the system date - the program will go through the following process:

1. Determine the Low-Level Code for each Item. This Code represents the lowest level in the Bill of Material structure that the Item appears and defines the priority sequence in which Items are processed.
2. The next step is to delete all previous Suggested Orders and Suggested Demands
3. Commencing with the first Item at Low Level Code 0 the Actual Demand is evaluated on a daily basis.
4. The first day's demand is assessed by adding any overdue Supply Orders to the Current Stock Level.
5. Is the 'expected' Stock Level below the Item's Re-Order Level?

No - Go to activity 6

Yes - Determine the greater of (a) the quantity required to bring it up to the Re-Order Level and (b) the Re-Order Quantity. Round this up in accordance with the Item's Order Multiple and create a Suggested Supply Order.

If this Item is 'Assembled' then get the component requirements from the Item's BOM and create 'Suggested Demand Orders' for each Component where the quantity is that required to make the parent 'Suggested Supply Order' quantity (including component

- scrap) and the required date is the Parent Item's required date, less its Leadtime.
6. Go to the next day.
 - If the MRP Horizon has been reached then select the next Item and go back to Activity 4
 - If it is not beyond the MRP Horizon then, commencing with the current stock level, add any Supply Orders (Assembly Orders or Purchase orders), and deduct any 'Cumulative Demand' scheduled for this day then go back to the Activity 5

The results of the Replenishment run are displayed via the '**Create Required Orders**' screen. What you will find is that, for the above Items, the following calculations have been carried out.

Item **A**. Current Stock is 0 therefore it needs to be brought up to its Re-Order Level using the Reorder Qty. Therefore a single Suggested Supply Order is created with the following information

Quantity = 4
 Required Date = today's date
 Start Date = Today minus 7 days Leadtime

Item **B**. The Suggested Supply Order to make Item **A** would generate a Suggested Demand Order for B to make A. Therefore a single Suggested Demand Order is created with the following information

Quantity = 8 (4 of Item A with a BOM requirement of 2 per A)

As the current Stock is 0 it needs to be brought up to its Re-Order Level of 1. Therefore the total demand is 9. This is satisfied by creating a Suggested Order using the minimum Reorder quantity held against B (10). Therefore a single Suggested Supply Order is created with the following information

Quantity = 10
 Required Date = today's date
 Start Date = Today minus 14 days Leadtime

There are many variables within this simple scenario such as

- The Sales, Job or Assembly Order demands come from the outstanding issue quantity against these orders and not the original Order quantity
- Similarly the Purchase or Assembly Order Supply comes from the outstanding receipt quantity and not the Order quantity

Try creating a Sales Order for **A** and schedule it for (say) next month then re-run the MRP. See if you can predict what the results will be

4.4 Create Required Orders

The 'Create Required Orders' screen looks at Suggested Orders generated from:

- The above Replenishment Routines plus
- Items, Descriptors, and Catalogue Items whose 'Supply Method' is 'Source on Demand' and a demand was generated by the Sales, Assembly, or Job Order

This routine allows you to modify and/or approve the requirements before finally converting them into physical Orders. When converting to physical orders you have the option to:

- Select the lines to convert to Order(s)
- Combine Lines (segregated by Supplier) into a single Purchase Order if Purchased.
- Combine Lines into a single Assembly Order if Assembled.

Go into **Inventory>Create Required Orders** and amend the run date to include orders up to (say)

then end of next month then click the 'OK' button. Note: The program will extract all the Suggested Orders whose **Required Date** less **Leadtime Days** is earlier than this date.

On the presented screen you will see all the Required Orders that satisfy the selection criteria. If you scroll to the left you will see the source of the demand. You can now:

- Select any line and - in the Detail tab - amend the Order quantity
- 'Check' the line(s) that are to be converted into Orders
- 'Check' the Radio Button to state whether you want a combined Order by Supplier or that each line will have a separate Order
- Click the '**Generate Orders for selected Requirements**' button

Go into **Purchasing>Purchase Orders** to view the generated Purchase Orders. Highlight the last Purchase Order in the 'List' view and click on the 'Lines' tab. In that view click on the '**Line Allocations**' tab you can see that it is linked to the source of the demand.

4.5 Order Inventory Availability

The Order Availability function in Ostendo assesses current stock levels to see which orders can be satisfied in part or in full. Additional option is given when receiving Purchased goods to carry out this assessment taking into account the current receipt. To begin this session go into **Requirements>Order Inventory Availability** and enter the following parameters

1. Run Parameter options

1.1. Order Priorities

The first step is to define the priority sequence of Orders. Allocations will be made in accordance with this priority. You can identify up to three priority levels. The first priority Level is the main priority sequence; the second is the priority within the first sequence, and so on. The available options against each Priority level are:

- None** - No priority sequence
- Required Date** - The Order Required Date
- Order Date** - The Date the Order was raised
- Customer Priority** - The Priority (1 to 9) defined against the Customer master
- Order Priority** - The Priority Number (1 to 9) defined against the Order
- Order Started** - Orders whose status is 'In Progress' come first

Select your Order Priority options.

1.2. Orders and Status Inclusions

In addition to defining the Priority you can also restrict/include Sales Orders, Job Orders, or Assembly Orders. By default all the Orders with status Open and In-Progress are included but you can also include Planned, On Hold, or Quote.

Select which Orders and their status' that you wish to include in this Order Allocation run

1.3. How to fill Orders

The next step is to define how you want to process available stock against the above selected orders. Three options are available

- Partial Fulfillments**: For each stock Item progressively allocate where there is a demand until all demands are satisfied or available stock has been consumed.
- Full Order Fulfilment** : All lines in the Order are checked to see if any line in the order cannot be fully satisfied. If so then no lines in the Order will be allocated stock
- Order Full Delivery Flag** - This is combination of the previous two in that if the Order is

flagged as 'Full Delivery' and any line cannot be fully satisfied then none of the lines in that Order will be included. Orders not flagged as 'Full Delivery' will be treated as 'Partial Fulfilments'

Select your preferred option

1.4. Other Parameters

Demand Up To Required Date: This denotes that all selected orders whose Required Date is equal to, or earlier than, this date will be included.

Restrict to Site: You can restrict the Order Availability run to all stock at a given Site.

Restrict to Purchase receipt: This option allows you to focus only on those Items in a specific Purchase receipt (in addition to current stock levels for those Items).

2. The Process

This describes the process that Ostendo will go through to determine Inventory Availability.

Firstly, Ostendo will generate a list of all included Orders up to the 'Demand Up to Required Date' parameter and sort them into the required priority sequence.

The first Order in the prioritised list is addressed as follows.

If the Fulfilment style is '**Partial Fulfilments**' then the first Item in the Order is matched against the available stock and either the demand is satisfied or the available stock has been consumed. The available stock against the Item record is reduced by this amount in preparation for matching against the next demand. The allocations continue with the next Line in the Order until all lines have been addressed.

If the Fulfilment style is '**Full Order Fulfilment**' then all Items in the Order are checked to see if there is sufficient stock to fully satisfy all lines. If:

- There is sufficient stock then the allocations are made. The available stock against each Item record is reduced by this amount in preparation for matching against the next demand.
- There is insufficient stock then no allocations are made against any line and the process continues with the next Order in the priority list.

If the Fulfilment style is '**Order Full Delivery Flag**' then the Order is checked to see if the 'Full Delivery' flag is checked and:

- If it is 'checked' then the allocation conforms to the 'Full Order Fulfilment' process
- If it is not 'checked' then the allocation conforms to the 'Partial Fulfilment' process

The next Order in priority sequence is then addressed in the same way until all orders have been processed.

3. The Results

A results screen is presented which provides information about the Order Allocation run. You should note that these evaluations are recommendations only and no update of the database has taken place.

3.1. Availability Results

The front screen displays the results of the Allocations in the same priority sequence that the process was carried out. Each line in this screen represents an individual Order. Most of the

fields in each line are display only. However two fields are prefilled with the results of the Order Allocation run and you have the option to make changes and re-run the Order Allocation process if required. These fields are:

Order Priority: This is the current Order Header priority. You can change this against any line and rerun the Order Allocation process. Note: This change will also update the current Priority against the Order Header.

No Shortages Allowed: This checkbox shows whether the Order allows shortages or if it must be 'Full Delivery Only'. You have the option to 'check' or 'uncheck' this box before rerunning the Order Allocation routine

3.2. Other Panels

Order Lines: You can select a line from the above 'Availability List' panel and drill down to look at the content of the Order.

Shortages: Shows all Items where there is insufficient stock to satisfy the Order demands.

4. Change Order Details

Having identified which Orders can be progressed further you now have the option to carry out the following actions against these.

4.1. Order Status Change

On the Toolbar (just above the detail grid) you will see a 'Release Orders' button. If you click this button then a drop-down will be presented from which you can convert the following orders:

- **Jobs:** Convert Job Orders from current **Open** status to **InProgress**
- **Assemblies:** Convert Assembly Orders from current **Open** status to **InProgress**
- **Delivery Sales:** Create a **Delivery** for this Order and change status to **InProgress**
- **Counter Sales:** Convert Sales Orders from the current **Open** status to **InProgress**

Make your selection and you will be asked to confirm this before the selected orders are converted

4.2. Document Print

On the Toolbar (just above the detail grid and to the right of the 'Release Orders' button) you will see a 'Print' button. If you click this button then a drop-down will be presented from which you can print documents for the Orders to be progressed. Make a selection from the following:

- **Job Sheets - Not Printed:** Only print Job Sheets that have not yet been printed
- **Job Sheets - All:** Print all selected Job Sheets
- **Assembly Sheets - Not Printed:** Only print Assembly Sheets that have not yet been printed
- **Assembly Sheets - All:** Print all selected Assembly Sheets
- **Delivery Pick Lists - Not Printed:** Only print Delivery Pick Lists that have not yet been printed
- **Delivery Pick Lists - All:** Print all selected Delivery Pick Lists.
- **Counter Pick Lists:** Print all selected Counter Sales Order Pick Lists.

Make your selection and you will be asked to confirm this before the selected documents are printed.

4.6 Inventory Changes

Within Ostendo there are 3 types of changes allowed

- **Inventory Change** - Amend current stock's sub-level variants
- **Base Unit Swap** - Allows you to change the Item's base Unit of Measure
- **Inventory Unit Change** - change an Item's Unit of Measure to another valid Unit held against the Item record

1. Inventory Change

This feature allows you to address the current sub-level variations (Serial Number, Batch Number, Colour, etc) of Items in stock and amend them as required without creating a Stock Movement Transaction. You should note that this feature does not allow changes to the stock Quantity, Warehouse, Location, or the Unit of Measure.

Step 1: Go into **Inventory>Items** and go to the 'Details' tab of Item **OC-7452** (Office Chair - Standard Gas). You will notice that the 'Colour' checkbox is 'checked'. If you click on the 'Colours' button to the right of the checkbox you will see that this Item has 4 variants. We will receive 10 'Red' Chairs into stock and then realise that they should have been 'Blue'. We will use this process to carry out that amendment.

Step 2: Go into **Inventory>Inventory Adjustments** and 'Add' a new batch. After saving the batch click on the 'Lines' tab and carry out the following:

- Click on the 'Add' button to create a new line record
- Click on the drop-down against 'Item Code' and select Item **OC-7452**.
- Enter an Adjustment Quantity of **10**
- Select the **Adjustment Type** 'Receipt'
- Select 'Red' from the drop-down list against 'Colour'
- Click the 'Save' button
- Go back to the 'Details' tab and click the 'Post all Adjustments' button

Step 3: Go into **Inventory>Inventory Change** and select **OC-7452** then:

- Click on the 'Detail' tab
- On the displayed record click the drop-down list under field 'Colour' and select 'Blue'
- Save the record
- If you now go to the Item Availability screen **Inventory>Inventory Availability** for **OC-7452** you will see that the Colour is now 'Blue'

2. Base Unit Swap

This feature allows you to adjust an Item's base Unit of Measure to another base unit. In this exercise you will replace the base unit held against Item **1500-2185** (Green Paint) from 'Litre' to 'Galls'.

Step 1: To commence this you first need to create the new Unit of Measure. Therefore go into **General>Settings>Standard Units** and create a new Unit called 'Galls'.

Step 2: Go into **Inventory>Items** and highlight Item **1500-2185**

- Click on the 'Related' button to the right of the screen and select 'Item Units' from the list
- In the displayed panel add a record showing
 - To Unit:** Select **Galls** from the drop-down list
 - Conversion Factor:** MUST be **1**
- Click the 'Save' Button

Step 3: Go into **File>System Configuration>Item Base Unit Swap**. In the presented screen:

- Select **Item Code 1500-2185**
- Select the new Base Unit '**Galls**'
- Click the '**Apply Change to entire Database**' button

Step 4: Finally, remove the 'old' Unit from the Item's list as follows

- Go into **Inventory>Items** and highlight Item **1500-2185**
- Click on the '**Related**' button and select '**Item Units**' from the list
- In the displayed panel delete the '**Litre**' line
- Click the '**Save**' Button

3. Inventory Unit Change

This feature allows you to adjust the Unit of Measure of an Item in Stock to another valid Unit of Measure applicable to the Item. In this exercise you will add an additional Unit of Measure against Item **PE-7721** (Standard Ballpoint Pen) of '**Box**' (**50** Pens per Box). We will then receive **5** 'Box' (es) into stock and then 'split' one box into individual Pens:

Step 1: Identify the Alternate Unit of Measure

- Go into **Inventory>Items** and select **PE-7721**
- Click on the '**Related**' button to the right of the screen and select '**Item Units**' from the list
- You will see that the alternate unit '**Box**' has a conversion Factor of **50**.

Step 2: Go into **Inventory>Inventory Adjustments** and '**Add**' a new batch. After saving the batch click on the '**Lines**' tab and carry out the following:

- Click on the '**Add**' button to create a new line record
- Click on the drop-down against '**Item Code**' and select Item **PE-7721**.
- Select '**Box**' from the drop-down list under '**Unit**'
- Select the **Adjustment Type** '**Receipt**'
- Enter an **Adjustment Quantity** of **5**
- Select '**Blue**' from the drop-down list under '**Colour**'
- Click the '**Save**' button
- Go back to the '**Details**' tab and click the '**Post all Adjustments**' button

Step 3: Go into **Inventory>Inventory Unit Change** and locate Item **PE-7721** then click on the '**Detail**' tab. Then:

- Select the line showing the **5 'Boxes'**
- Click on the '**Change Unit of Measure for selected Record**' button
- Leave the '**From Qty**' as **1**
- Select '**Each**' from the drop down list against '**To Unit**'
- Click the '**Update**' button
- If you now go to the Item Availability screen **Inventory>Inventory Availability** for **PE-7721** you will see that the stock quantity against **5 Boxes** has reduced to **4** but the stock quantity against **Each** has increased by **50**.

4.7 Assembly Order Backflushing

Backflushing is used during Assembly Order receipts to automatically issue components required to make the Assembled Item at the time the Item is received into stock.

1. Preparation

You can define if you want all Assembly receipts to automatically backflush or only have this feature against selected Items.

To set the Backflush to be all receipts go to **Inventory>Settings>Item Rules** and select 'All Items' under drop-down against field 'Assembly Backflush Policy'.

In this training session we will backflush against a specific Item. Therefore:

- Go into **Inventory>Settings>Item Rules** and select 'Item Specific' under drop-down against field 'Assembly Backflush Policy'.
- To identify the specific Item go into **Inventory>Items** and select Item **1105-2184** (Handle Assembly). In the 'Detail' screen click on the 'Additional Inventory Settings' button. In that panel 'check' the 'Backflush Issues on Assembly Receipts' checkbox

2. Backflushing in action

2.1. Order Creation

Go into **Assembly>Assembly Orders** and create an Assembly Order for **40** off **1105-2184**

If you click on the 'Lines' tab within the Order you will see that the BOM was copied with the following amounts

Item Code **760-2176** (Tube-Stainless Steel-25mm1200mm) **40** off
Labour Code **LAB-ASSEMBLY** (Assembly Labour) **40** hours
Item Code **900-2182** (Handle Grip-Rubber-25mm) **80** off

Add the following line to the Order Line

Descriptor Code **GENERALTIME** (Time - Non Employee Related) **10** hrs

Note the Order Number then close out of the Assembly Order screen

2.2. Receive and Backflush

Go to **Assembly>Assembly Receipts** and create a Receipt batch. Click on the 'Lines' tab and - in that screen - click on the 'Select Assembly for prefilling receipts' button. On the displayed panel select the above Assembly Order. This should create a single line displaying Item **1105-2184**. Scroll to the 'Qty' field and change the prefilled quantity to **10**. Save the record then click on the 'Detail' tab. On the Detail screen click on the 'Post all Receipt Entries' button. Then close out of the Assembly receipts screen.

Go back to **Assembly>Assembly Orders** and select the above Order and click on the 'Lines' tab. You will see that it has backflushed the Item Codes and the Descriptor but NOT the Labour Code. The reason for this is that Employee Times booked through the Timesheet (**Labour>Timesheets**) will be used to update Labour Activities. Having said this you may not wish to use the Timesheet function but simply get Ostendo to 'Backflush' Labour Code activities. We will do this in the next step.

2.3. Backflush Labour Codes

Go into **Assembly>Settings>Assembly Rules** and 'check' the checkbox 'Include Labour When Backflushing'. Once 'checked' you should also nominate a default Employee (say **Bob Drum**) from the drop-down against 'Default Employee for Backflushing'.

Now go to **Assembly>Assembly Receipts** and create a Receipt batch. Click on the 'Lines' tab and - in that screen - click on the 'Select Assembly for prefilling receipts' button. On the displayed panel select the above Assembly Order. As before scroll to the 'Qty' field and change the prefilled quantity to **10**. Save the record then click on the 'Detail' tab. On the Detail screen click on the 'Post all Receipt Entries' button. Then close out of the Assembly receipts screen.

Go back to **Assembly>Assembly Orders** and select the above Order and click on the 'Lines' tab. You will see that it has backflushed the Labour Code in addition to the Item and Descriptor Codes. You may wish to click on the 'Actual Issues' tab in the lower part of this screen to look at the Issue History for the selected line.

2.4. Items with variants

If the Item has sub-level variants (such as Colour, Serial Number, Batch, etc) then the backflush cannot automatically take place. In this instance an 'Issue Batch' is generated against which you should manually state which variant has been issued. Let's see how this works:

Go back to **Assembly>Assembly Orders** and select the above Order and click on the 'Lines' tab. Add a new line containing the following Item that has 3 variations of Colour

Item Code **OC-7451** (Office Chair - Fabric Executive) **40** off

Go back into **Assembly>Assembly Receipts** and create another Receipt batch. Click on the 'Lines' tab and - in that screen - click on the 'Select Assembly for prefilling receipts' button. On the displayed panel select the above Assembly Order. As before scroll to the 'Qty' field and change the pre-filled quantity to **10**. Save the record then click on the 'Detail' tab. On the Detail screen click on the 'Post all Receipt Entries' button. Then close out of the Assembly receipts screen.

If you go back to **Assembly>Assembly Orders** and select the above Order and click on the 'Lines' tab. You will see that Backflushing has not been carried out against **OC-7451**. However, if you go to **Assembly>Assembly Issues** you will see that an Issue Batch has been created to cover this. That Issue Batch will require the specific variant to be selected before it is issued

2.5. Source On Demand

If the Assembly Order Component Line is an Item Code with 'Supply Method' of 'Source on Demand' then the Item will not be backflushed because a supply Order is scheduled to be generated to satisfy this demand

Similarly, if the Assembly Order Component Line is a Descriptor with 'Source By' of 'Purchasing' then the same condition will apply

5 4. Sell Price Buy Price and Costs

Ostendo contains a very flexible and comprehensive pricing and costing structure covering

Sell Prices: The price at which you sell a product or service

Buy Price: The price at which you Purchase a product or service

Cost: The cost of the product or service (This can be Assembly Order Costs or Purchase Buy Price plus other costs)

5.1 Sell Price Maintenance and Use

The Sell Price structure covers the following areas:

- Inventory Items
- Non-Inventory Items (In Ostendo these are called 'Descriptors')
- Labour Charges
- Kit Sets (Kits of Parts)
- Tasks (Service oriented activities)
- Supplier Catalogues

The following exercises cover the determination and use of the Sell Price, which includes Pricing options such as:

- Basic Sell Price
- Price Matrix including quantity Discount Pricing
- Unit of Measure Pricing
- Special (or Promotional) Pricing
- 'On Sell' Pricing

1. Inventory Items

Basic Sell Price

A Base Sell Price is held against each Inventory Item. To add the Base Sell Price go into **Inventory>Items** and select Item **1500-2185** (Green Paint). You will see that it already has a Sell Price in the '**Standard Sell Price**' field. Amend this if required. In the absence of any other Sell Price conditions (defined below) this price will be used. If no Price is entered here then a zero Base Price is assumed.

Sell Price Matrix

A Sell Price matrix can be established against each Item where a Base Sell Price and Quantity Discounts can be defined for user-defined Price Levels. A single Price Level is linked to a Customer record and whenever an Order is raised the Customer's Price Level price is used. Let's see how this works.

- You should first set up Price levels by going into **Pricing>Settings>Price Levels**. Try adding a couple of Price levels such as '**Trade**' or '**Wholesale**'
- The next step is to create the Prices by Price Level. Go into **Pricing>Item Pricing** (or click on the '**Pricing**' button on the Item Screen) and create a Sell price with Quantity Breaks for the Price Levels you have just created.
- Finally, go into the Customer Screen (**Sales>Customers**) and select a Customer. Click on the '**Pricing and Invoicing**' tab and attach one of the above Price Levels to the Customer
- Now create a Sales Order for the Customer by going into **Sales>Sales Orders**. '**Add**' the above Item to the Order Line. You should see the Price-Level price for the Item/Quantity will come through to the Order Line

Unit of Measure Sell Price

If the Inventory Item has multiple Units of Measure then each UOM can offer a Sell Price based on

a percentage of the Base Price. For example, if the Items basic UOM is (say) 'Litre' and it is also stocked and sold in 'Case' then a UOM discount matrix can be set up as follows:-

Sell Unit	Sell Price Discount
Litre	0%
Case	15%

To create these discounts go back to **Inventory>Items** and select Item **1500-2185** (Green Paint).

Click on the 'Related' button down the right-hand side of the screen and select 'Item Units'. In the presented screen add the following record

To Unit: Case
Conversion Factor: 10 (I.e. 10 Litres per Case)
Sell Price Discount: 15%

Go back to the above Sales Order and add this Item to the Order Line. Amend the Unit from the base 'Litre' to the new unit 'Case'. You should see the Price-Level price for the Item/Quantity adjusted by the 15%.

Special Pricing

Special Pricing allows for the maintenance of promotional pricing of Items. To create a special price go into **Pricing>Special Pricing** and create a special Price - covering a brief period - for the above Item. Go back to the above Sales Order and add this Item to the Order Line. You should see the Special Price come through to the Order Line if the Date range of the Special Price is effective to the Order Line creation Date

'On Sell' Pricing

'Add-on' Sales cover those additional products or services you wish to promote when certain Items are added to a Sales Order. Example:- If the Customer orders Item **1500-2185** (Green Paint) you may also wish to promote Item **1500-2187** (Blue Paint) at the same time. Each 'Add-on' Item can have a single price specifically linked to this promotion or alternatively can use its standard pricing matrix.

To create an 'On Sell' Price go into **Inventory>Items** and select Item **1500-2185** (Green Paint) and go to the 'Detail' tab. 'Check' the 'Add-On Sales Apply' checkbox then click on the adjacent 'Item Add-On Sales' button. Add the details of **1500-2187** (Blue Paint) into the presented screen and enter a Special Price to be used when this is sold as 'Add-on' Sale Item.

Go back to the above Sales Order and add Item **1500-2185** (Green Paint) to the Order Line. You should see an 'Add-On Sale' button appearing in the 'Line Info' Bar. If you click on this button then the 'Add-On' Sales details will be displayed. You can then offer **1500-2187** (Blue Paint) which, if accepted, will be included in the Sales Order Line with the associated Special Price.

Job Order 'Markup' Price

This type of Sell Price is specific to Job Orders and only applies where the Item in the Job Order Line has been specifically ordered for the Job. In this instance the Purchase Price will be used as a basis for evaluating the Sell Price. This will be covered in detail in the Job Order Training guide. However, you can see the setup options by going into **Inventory>Items** and selecting Item **1500-2185** (Green Paint). You will see two fields;

Checkbox: 'Calculate Sell Price from Receipt Cost for Jobs'
Field: Markup

2. Descriptors

A Base Sell Price is held against each Descriptor. To add the Base Sell Price go into **Inventory>Descriptors** and 'Add' a new Descriptor with the following information

Descriptor Code: Subcontract
Unit: Hours

Description: Sub Contract Labour

Enter a Sell Price into field '**Standard Sell Price**'. In the absence of any other conditions (defined below) this price will be used. If no Price is entered then a zero Base Price is assumed.

Sell Price Matrix

A Sell Price matrix can be established against each Descriptor where a Base Sell Price and Quantity Discounts can be defined for user-defined Price Levels. A single Price Level is linked to a Customer record and whenever an Order is raised the Customer's Price Level price is used.

- You should first set up Price levels by going into **Pricing>Settings>Price Levels**. Try adding a couple of Price levels such as '**Trade**' or "**Wholesale**' if you have not already created from the exercise above
- The next step is to create the Prices by Price Level. Go into **Pricing>Descriptor Pricing** (or click on the '**Pricing**' button on the Descriptor Screen) and create a Sell price with Quantity Breaks for the Price Levels you have just created.
- Finally, go into the Customer Screen (**Sales>Customers**) and select a Customer. Click on the '**Pricing and Invoicing**' tab and attach one of the above Price Levels to the Customer
- Now create a Sales Order for the Customer by going into **Sales>Sales Orders**. Add this Descriptor to the Order Line. You should see the Price-Level price for the Descriptor/Quantity come through to the Order Line

Special Pricing

Special Pricing allows for the maintenance of promotional pricing of Descriptors. To create a special price go into **Pricing>Special Pricing** and create a special Price - covering a brief period - for the above Descriptor. Go back to the above Sales Order and add this Descriptor to the Order Line. You should see the Special Price come through to the Order Line if the Date range of the Special Price is effective to the Order Line creation Date.

Job Order 'Markup' Price

This type of Sell Price is specific to Job Orders and only applies where the Descriptor in the Job Order Line has been specifically ordered for the Job. In this instance the Purchase Price will be used as a basis for evaluating the Sell Price. This will be covered in detail in the Job Order Training guide. However, you can see the setup options by going into **Inventory>Descriptors** and selecting Descriptor Subcontract. On that screen you will see two fields;

Checkbox: '**Calculate Sell Price from Receipt Cost for Jobs**'

Field: **Markup**

3. Labour Codes

You can charge Labour at two levels within Ostendo:

- If you only wish to identify and charge Labour to an Assembly Order or Job Order and subsequently enter the actual Labour hours used then it is easier to identify the Labour as a Descriptor and book actual quantities against the Descriptor.
- If you wish to identify the **Labour Code** against an Assembly Order or Job Order and subsequently have a specific **Employee** book time against that Labour Activity then you must use the Labour Code process. In this instance the Labour Code will provide the Planned Charge Rate and the Employee the Actual Charge Rate.

Basic Charge Rate

A Base Charge Rate is held against each **Labour Code**. To add the Base Charge Rate go into **Labour>Labour Codes** and select **LAB-GENERAL**. You will see that a charge rate has already been entered into field '**Standard Rate**'. In the absence of any other conditions (defined below) this Charge Rate will be used. If no Charge Rate is entered then zero is assumed.

Charge Rate Matrix

A Charge Rate matrix can be established against each Labour Code where different Charge Rates can be defined for user-defined Rate Levels. A single Rate Level is linked to a Customer record and whenever an Order is raised the Customer's Rate Level price is used. To see how this works:

- You should first set up **Rate Levels** by going into **Labour>Settings>Rate Levels**. Add a couple of Rate levels such as '**Trade**' or '**Wholesale**'
- The next step is to create the Charge Rate by Rate Level. Go into **Labour>Rate Levels** and create a **Charge Rate** for the **Rate Level(s)** you have just created.
- Finally, go into the Customer Screen (**Sales>Customers**) and select a Customer. Go to the '**Pricing and Invoicing**' tab then, in the drop-down against field '**Rate Level**' select the level that will apply to this Customer
- Now create a Sales Order for the Customer by going into **Sales>Sales Orders**. '**Add**' this **Labour Code** to the Order Line. You should see the Rate-Level price for the **Labour Code** come through to the Order Line

4. Discount Matrix

The above three sections calculate the Sell Price. You can optionally apply a Discount to the calculated price. This provides the following options

- Sell Price Calculation only using the above Sell Price Matrices
- Discount only using the Item, Descriptor or Labour Code's Base Price
- Discount the Calculated Sell Price
- No Sell Price Matrix or Discount - just use the Base Price

To create a Discount go into **Pricing>Discount Matrix** where you will see the Discounting maintenance screen. In this exercise we will give a Discount to Customer '**Jim Gold & Co Ltd**' when they purchase Item Code **1500-2186** (Red Paint). Click on the '**Add**' button to create a new Discount record and enter the following into the respective fields

Discount Level: Customer
Discount Level Name: Jim Gold & Co Ltd
Product Level: Item
Product Level Code: 1500-2186
Discount: 25%

If you now create a Sales Order (**Sales>SalesOrders**) for **Jim Gold & Co Ltd** then, in the '**Lines**' tab add Item **1500-2186** you will see that the **25%** Discount has been applied to the Sell Price

5. Kit Set Pricing

A Kitset is a collection of Items and Descriptors that are sold as a 'single entity'. The Kitset itself has its own Sell Price whereas the cost of the Kitset comes from the individual Items and Descriptors issued to the Kit.

Kitset Price

To create a Kitset simply create a **Descriptor** as described in 1.2 and - on the Descriptor's '**Detail**' screen - 'check' the '**The Descriptor is Used as a Kitset Code**' Radio Button. The Sell Price and Sell Price Matrix of the Descriptor (as described in 1.2.) will now define the Sell Price of the Kitset.

Kitset Components

The components used within a Kitset are maintained via **Sales>Kitsets**. Go into that screen and

select the Descriptor you have just created. Go into the 'Lines' tab of the Kitset and add a couple of Items and/or Descriptors to the Kitset.

Kitset Price Maintenance

If you now click on the 'Detail' tab of the Kitset you will see the Sell Price that comes from the Descriptor record. You can amend this sell price in this screen by 'checking' the appropriate 'Radio Button' and saving the change. The sell price, therefore, can come from one of three sources:-

- Price as (manually) entered in the Kitset's Descriptor record, or
- Rolled-up from the Base Sell Price of the Kit's Components
- Rolled-up from the Base Cost of the Kit's Components multiplied by a Markup %

It should be noted that, for the last two options, whenever a change is made to any of the Kitset's Item or Descriptor's Base Sell Price (or Cost) then the program will automatically update the Base Kitset Price against the 'Kitset' Descriptor record.

Sales Order Kitset Price

Now create a Sales Order for the Customer by going into **Sales>Sales Orders**. Go into Sales Order 'Lines' tab and click the 'Add' button. Select 'Line Type' of 'Kitset Code' then select the above Kitset in field 'Code'. You should see the Price-Level price for the Kitset Code come through to the Order Line

Special Pricing

Special Pricing allows for the maintenance of promotional pricing of Kitsets. To create a special price go into **Pricing>Special Pricing** and create a special Price - covering a brief period - for the above 'Kitset' Descriptor. Go back to the above Sales Order and add this 'Kitset' to the Order Line. You should see the Special Price come through to the Order Line if the Date range of the Special Price is effective to the Order Line creation Date.

6. Task Bill Pricing

A Task Bill is a collection of Items, Descriptors and Labour activities that are assembled to perform a defined activity. This activity contains a standard 'Sell Price' for quoting and planning purposes.

Task Bill Price

To create a Task Bill simply create a Descriptor as described in 1.2 and - on the Descriptor's 'Detail' screen - 'check' the 'The Descriptor is Used as a Task Bill Code' Radio Button. The Sell Price and Sell Price Matrix of the Descriptor (as described in 1.2.) is used to define the Sell Price of the Task Bill.

Task Bill Content

The content of the Task Bill is maintained via **Jobs>Task Bills**. Go into that screen and select the Descriptor you have just created. Go into the 'Lines' tab of the Task Bill and add a couple of Items/Descriptors/Labour Codes.

Task Bill Price Maintenance

If you now click on the 'Detail' tab of the Task Bill you will see the Sell Price that comes from the Descriptor record. You can amend this sell price in this screen by 'checking the appropriate 'Radio Button' and saving the change. The sell price, therefore, can come from one of three sources:-

- Price as (manually) entered in the Task Bill's Descriptor record, or
- Rolled-up from the Base Sell Price of the Task Bill's Components
- Rolled-up from the Base Cost of the Task Bill's Components multiplied by a Markup %

It should be noted that, for the last two options, whenever a change is made to any of the Task Bill'

s content then the program will automatically update the Base Task Bill Price against the 'Task Bill' Descriptor record.

Job Order Task Bill Price

Now create a Job Order for a Customer by going into **Jobs>Job Orders** and clicking the 'Add' button. In the 'Job Order Creation' panel select a Customer then click on the 'Task Information' tab at the top of the panel (i.e. Do not generate the Job Order yet). Under field 'How Job Lines are related to this Task' select 'A Single Task Bill is linked to this Task'. Then in the field 'Task Bill Code' select the above Task Bill from the drop-down list. Finally click on the 'Create Order' button.

If you go into the 'Lines' tab of the generated Job Order you will see the Price-Level price for the Task Bill Code comes through to the Order Line. This whole Job Order process is covered in detail in the Job Orders Training Guide

Special Pricing

Special Pricing allows for the maintenance of promotional pricing of Task Bills. To create a special price go into **Pricing>Special Pricing** and create a special Price - covering a brief period - for the above 'Task Bill' Descriptor. If you create a Job Order as described in the previous paragraph you will see the Special Price come through to the Order Line if the Date range of the Special Price is effective to the Order Line creation Date.

7. Supplier Catalogue Item Pricing

A simple Import Routine allows the user to create and maintain multiple Supplier Catalogues. These catalogues can contain both a Buy Price and a Recommended Sell Price along with a Sell Price Matrix.

If you go to **Purchasing>Supplier Catalogues** you will see that a catalogue already exists for Supplier 'Electrical Power Company'. If you click on the 'Lines' tab you will see that each record contains a Buy Price and also the Supplier's Recommended Sell Price.

Catalogue Item Sell Price Matrix

A Sell Price matrix can be established relating to all Items in the Catalogue based on the Supplier's Buy Price or Recommended Sell Price. To do this:

- You should first set up Price levels by going into **Pricing>Settings>Price Levels**. Try adding a couple of Price levels such as 'Trade' or 'Wholesale' if you haven't already done so.
- The next step is to create the Prices by Price Level. Go into **Purchasing>Supplier Catalogues** and select the 'Detail' tab. In the lower part of the screen select the 'Pricing Level' tab and define the global Sell Price variable that will apply to the Price Level you have just created.
- Finally, go into the Customer Screen (**Sales>Customers**) and select a Customer. Click on the 'Pricing and Invoicing' tab and attach one of the above Price Levels to the Customer
- Now create a Sales Order for the Customer by going into **Sales>Sales Orders**. Go to the Order 'Lines' and, on the 'Batch Entry' bar select the 'Catalogue Items' button. From there select an Item and you will see that the Sell price is derived from the Base Catalogue Price taking into account the Price Level variable.

8. Customer Discounts

In addition to the above pricing you can also offer a discount by Customer that applies to the whole Sales Invoice. The percentage discount is initially stored against the Customer Record and copied

to the Sales Order where it can be amended and made specific to that order. This Discount appears on the Invoice in its own right as a Customer Order Discount.

To create a Customer Discount go into **Sales>Customers** and select a Customer. Click on the '**Pricing and Invoicing**' tab and enter a Discount percentage into field '**Discount%**'. To show how this works do the following:

- Go into **Sales>Sales Orders** and generate an Order for the above Customer
- In the '**Lines**' tab click on the '**Descriptors**' Button on the '**Batch Entry**' bar and add '**General Time**' with (say) **10** Hours.
- Click on the '**Picking**' button in the '**Batch Entry**' bar and 'Pick' the Line quantity.
- Click on the '**Update Sales Order**' button then close out of the screen
- Return to the Sales Order '**Detail**' tab
- You will see that the discount percentage and calculated discount amount will be displayed in the lower left of that screen

9. Price Updates

A simple routine enables you to maintain Item and Descriptor Sell prices. It comprises of selecting a range of Items or Descriptors and applying a % change (plus or minus) based on Standard Sell Price, Standard Buy Price, or Last Cost.

To adjust the prices go into **Pricing>Batch Price Update** and try updating some Items and/or Descriptors

10. Price Inquiries

Quite often a Prospect or Customer may wish to know the current sell price. Go into (**Pricing>Price Inquiry**) and enter the following

- Customer Name (I.e. Price Level known) or Price level
- Item or Descriptor identity (select from drop-down list)
- Quantity required (to take advantage of any Quantity Price Breaks)
- Unit of Measure (Base unit defaulted - amend from drop-down) for UOM discounts

Upon entering this information a display panel shows the Sell Price and all the applied discounts that enabled the price to be calculated

5.2 Buy Price maintenance and use

Ostendo contains a comprehensive Supplier Buy pricing structure covering

- Inventory Items
- Non-Inventory Items (In Ostendo these are called 'Descriptors')
- Supplier Catalogues

This document covers the processing and use of Buy Pricing options such as:

- Standard Buy Price
- Supplier Specific Pricing
- Quantity Discounts
- Supplier Unit of Measure Pricing

These exercises take you through the way Buy prices are created and maintained. It should be noted that a Buy Price differs from a Cost in that the Cost could be the Buy Price PLUS additional charges such as Freight, Postage, etc.

1. Item Buy Price

Standard Buy Price

A Base Buy Price is held against each Inventory Item. To add the Base Buy Price go into **Inventory>Items** and select Item **100-2000** (Washer-Mild Steel-8mm). You will see that it already has a Buy Price in the **'Standard Buy Price'** field. Amend this if required. In the absence of any other Buy Price conditions (defined below) this price will be used. If no Price is entered then a Base Buy Price of zero is assumed.

Supplier-based Buy Price

Against each Item facility is provided to add Supplier-specific Buy prices. This price is used when ordering the Item from that Supplier. To create the Supplier Specific 'Buy Price' go into **Inventory>Items** and select Item **100-2000** (Washer-Mild Steel-8mm). In the **'Detail'** screen click on the **'Pricing'** Button (Alternatively you can go into **Pricing>Item Pricing** to go to the same record). In the Pricing screen click on the **'Buy Prices'** tab half way down the screen. Click the **'Add'** button and enter a new line as follows:

Supplier: Select the Supplier to which this Buy Price will apply.

Unit: This is prefilled with the base Unit of measure for the Item but can be amended to reflect the specific Unit of Measure of the Item from this Supplier.

Supplier Item Code: You can (optionally) identify the Supplier's Item Number if it is different the above Item Number. Both numbers will appear on the Purchase Order.

Unit Price: Enter the Buy Price relative to the Supplier's Unit entered in this line.

Quantity Breaks: You can enter up to 5 Quantity Breaks

Now let's see how this works:

- Go into **Purchasing>Purchase Orders** and click the **'Add'** button. In the drop-down list in field **'Supplier'** select the above Supplier and click the **'Create'** button
- Go into the **'Lines'** tab and select **'Item Code'** in the drop-down under **'Line Type'** and then select Item **100-2000** from the drop-down under **'Code'**. You will see that the displayed price is that defined against this specific Supplier.
- Go back to the **'Detail'** tab and click on the **'Print'** button. You will see both your Item Number and the Supplier's Item number in the Purchase Order.

One point to note is that you can enter a Supplier's Item Number when creating a Purchase Order Line. Let's see how this is done:

- Go to **File>System Configuration>System Settings** and make sure that the **'Advanced Searching'** checkbox is 'checked'
- Go back to the above Purchase Order **'Line'** screen and **'Add'** a new line. Click on the drop-down icon against field **'Code'** and - in the displayed panel - enter the Supplier's Item Number (or even a partial number) in the **'Search'** field at the bottom of the screen. This should show you the Supplier's Item Number being linked to your own Item Number

Restrict to defined Suppliers

If you go into **Pricing>Item Pricing** for **100-2000** and click on the **'Buy Prices'** tab you will see a checkbox that can be set to restrict purchases from Suppliers in the Buy Price List. Any attempt to purchase this Item from other Suppliers will be rejected.

Primary Supplier

Against any Item you can identify a Primary Supplier. This is used during automatic ordering via the Replenishment run and prefills the Purchase Order's Supplier with this Primary (or preferred) Supplier. Go into **Inventory>Items** and select **100-2000**. On the **'Detail'** screen select the primary (or preferred) Supplier from the drop-down against field **'Primary Supplier'**

2. Descriptor Buy Price

Standard Buy Price

A Base Buy Price is held against individual Descriptors. To add a Base Buy Price go into **Inventory>Descriptors** and select Descriptor '**SubContract**' (the one that you created above). Add a Buy Price in field '**Standard Buy Price**'. In the absence of any other Buy Price conditions (defined below) this price will be used. If no Price is entered then a Base Buy Price of zero is assumed.

Supplier-based Buy Price

Against each Descriptor facility is provided to add Supplier-specific Buy prices. This price is used when ordering the Descriptor from that Supplier. To create the Supplier Specific 'Buy Price' go into **Inventory>Descriptors** and select Descriptor '**SubContract**'. In the '**Detail**' screen click on the '**Pricing**' Button (Alternatively you can go into **Pricing>Descriptor Pricing** to go to the same record). In the Pricing screen click on the '**Buy Prices**' tab across the centre of the screen. Click the '**Add**' button and enter a new line as follows:

Supplier: Select the Supplier to which this Buy Price will apply.

Unit: This is prefilled with the base Unit of measure for the Descriptor but can be amended to reflect the specific Unit of Measure from this Supplier.

Supplier Code: You can (optionally) identify the Supplier's equivalent Code if it is different to the above Descriptor Code. Both numbers will appear on the Purchase Order.

Unit Price: Enter the Buy Price relating to the above Supplier's Unit of Measure.

Quantity Breaks: You can enter up to 5 Quantity Breaks

Now let's see how this works:

- Go into **Purchasing>Purchase Orders** and click the '**Add**' button. In the drop-down list in field '**Supplier**' select the above Supplier and click the '**Create**' button
- Go into the '**Lines**' tab and select '**Descriptor Code**' in the drop-down under '**Line Type**' and select Descriptor '**SubContract**' from the drop-down under '**Code**'. You will see that the price is that defined against this specific Supplier.
- Go back to the '**Detail**' tab and click on the '**Print**' button. You will see both your Descriptor Code and the Supplier's equivalent Code in the Purchase Order.

One point to note is that you can enter a Supplier's Equivalent Code when creating a Purchase Order Line. Let's see how this is done:

- Go to **File>System Configuration>System Settings** and make sure that the '**Advanced Searching**' checkbox is 'checked'
- Go back to the above Purchase Order '**Line**' screen and '**Add**' a new line. Click on the drop-down icon against field '**Code**' and - in the displayed panel - enter the Supplier's Equivalent Code (or even a partial code) in the '**Search**' field. This should show you the Supplier's Code being linked to your own Descriptor Code.

Restrict to defined Suppliers

If you go into **Pricing>Descriptor Pricing** for '**SubContract**' and click on the '**Buy Prices**' tab you will see a checkbox that can be set to restrict purchase from Suppliers in the Buy Price List. Any attempt to purchase this Descriptor from other Suppliers will be rejected.

Primary Supplier

Against any Descriptor you can identify a Primary Supplier. This is used during automatic ordering via the 'Create Required Orders' routine and prefills the Purchase Order's supplier with this Primary (or preferred) Supplier. Go into **Inventory>Descriptor** and select **SubContract**. On the '**Detail**' screen select the primary (or preferred) Supplier from the drop-down against field '**Primary Supplier**'

3. Buy Price Updates

Two routines enable you to maintain Item and Descriptor Buy prices.

3.1. Batch Buy Price Update

This comprises of selecting a range of Items or Descriptors and applying a % change based on Standard Sell Price, Standard Buy Price, or Last Cost. To carry out the Buy Price Update go into **Pricing>Batch Price Update**. You can update either Items or Descriptors. Try updating the Buy Price against the Item or Descriptor you created above

3.2. Price Update from File

This function allows you to maintain standard Buy Prices or Sell Prices in some other database or spreadsheet and import these into Ostendo. This feature requires that the other database is capable of exporting to .csv or .xls format. The Price Update process uses a standard Ostendo routine to carry out this process and comprises the following Steps

- Create a Price Update Batch and define the update criteria
- Point the import function to the .csv or .xls file
- Match the import file to the fields in the .csv or .xls file
- Run the import function to a temporary file
- View the results and re-run if necessary
- Update the prices

To go through this process go to **Pricing>Price Update from File** and look at what is available. If you are going to go through this routine please refer to the User Reference Guide.

5.3 Costing

Before you create or maintain Cost information it is necessary to understand what is meant by 'Costing'.

1. Introduction

1.1. What are Costs?

Costs are used in two main areas:

Planned Cost: Where the cost is used to derive the projected Cost of the Sale, Job or Project. These are used for Planning (or Estimating) purposes to determine the expected profit Margin when compared against the projected Sell Price.

Actual Cost: Where the costs recorded are the true Cost of the Sale, Job or Project. This is used to:

- Compare against the Planned Cost to focus on what and why there are differences
- Compare against the Sell Price to determine actual Profit Margins

1.2. Cost Methods

The type of activity carried out within a Business generally governs the cost method(s) used. Here are brief descriptions of the types of Cost Methods and the environments where they are used

Standard Costs

Mainly used in Manufacturing Industries where, when receiving manufactured goods into stock, the Actual cost is not known. This may be due, for example, to the Labour cost only being available at the end of the week when the Employee's 'Weekly' Timesheet is entered. A Standard Cost is a pre-defined fixed cost covering a period of time and can be used against Items, Descriptors and Labour Codes

Average Costs

Only used with Inventoried Items. It is commonly used by Sales, Distribution, and Small Businesses and provides a more accurate assessment of Inventory valuation by using the 'Actual' receipt Cost as it's basis. In these environments restocking is almost always from Purchases and therefore the Actual Cost of the receipt is generally known at the time of receipt. Average Costs are maintained on an ongoing basis by adjusting the Inventory Average each time a Stock receipt is made. It uses the following formula

$$\text{Item Average Cost} = \frac{(\text{Current Ave} * \text{Current Qty}) + (\text{Received Qty} * \text{Received Cost})}{(\text{Current Qty} + \text{Received Qty})}$$

Whenever an Item is issued then the current Average Cost is used

Actual Costs

Primarily used by those businesses that calculate the Sell Price based upon the Cost (using a Markup or Margin percentage). Quite often used in the Service and Construction Industry. The Actual Cost can come from Items or Descriptors purchased specifically for the Job

Last Cost

Where a Purchase price can fluctuate (Example: commodities) then some businesses use the Last Purchase Cost as the basis for creating an Estimate or Quote.

Buy Price

Many computer systems assume that the Purchase Buy Price is also the Cost. This is not the case in Ostendo as it takes into account extra costs over and above the Buy Price (Example: Duty, Transportation, Shipping, etc) to arrive at a true Cost. However, some users still prefer to use the Buy Price as the Cost and provision to use this option is available in Ostendo.

Calculated

Some businesses sell a variable Product or Service that is not stocked but is made 'to-order' and the cost of that Product or Service is based upon the cost of ingredients and/or activities to make it. The Product/Service can be specific to each order and therefore its cost can be dynamically generated

2. Where and How Costs are Used

You can see from above that 'Costs' can be used in many ways depending upon a User's requirements. The following steps show you how to set up the various options. It is not the objective of this exercise to go through ALL costing scenarios but simply to show all the available options. You can then select the option that applies to you and address the derived costs in the Sales, Assembly, Job and Purchase Order processes.

2.1. Sales Order Costing

Sales Order Lines can include

- Item Codes
- Descriptor Codes
- Kitset Codes
- Catalogue Codes

- Custom Products (i.e. A product designed to a Customer's specific requirement)

For each of these you would determine Planned and Actual Costs from the following sources.

Planned Costs

Items: When an Item is added to a Sales Order the system-wide Costing Method defined for Inventory is used. Go into **File>System Configuration>System Settings** and select the method in field '**Inventory Costing Method**'.

Descriptors: When a Descriptor is added to a Sales Order the system-wide Costing Method defined for Descriptors is used. Go into **File>System Configuration>System Settings** and select the method in field '**Descriptor Costing Method**'.

Kitset: A Kitset is an accumulation of components (Items and Descriptors). The Planned Cost comes from the sum of the costs of the Kitset Components. Those Components use the Costing Methods you identified above. Have a look at a Kitset via **Sales>Kitsets**.

Catalogue Code: The Cost of Catalogue Code items is the '**Buy Price**' as held against the Catalogue record. You can see these by going into **Purchasing>Supplier Catalogues** and clicking on the '**Lines**' tab

Custom Product: The Cost of Custom Product is derived from functions within the Custom Product script. If the costing function in the script is not used then the Item/Descriptor/Catalogue Code costs described above will be used

Actual Costs

Items: An Item is always issued to a Sales Order from stock. Therefore the cost of the Item is determined by the '**Inventory Costing Method**' as defined in **File>System Configuration>System Settings**. However,

- If the Item (**Inventory>Items**) is Serial Numbered controlled and is flagged as '**Actual Costing is Used**' then the Actual Cost held against the issued Serial Numbered Item will be used.
- If the Item (**Inventory>Items**) is Batch Controlled and is flagged as '**Actual Costing is Used**' then the 'Batch Average' Cost held against the Item's Batch Number will be used.

Descriptors: The cost that will be applied will be related to the 'Descriptor Costing Method' as defined in **File>System Configuration>System Settings**.

Kitset: The Actual Cost of the Kitset is derived from the individual Kit Components that were actually issued to the Sales Order. As the components can be both Items and Descriptors then the above 'Actual' Cost methods will be used

Catalogue Code: The 'Actual' Cost of Catalogue Code items is the '**Buy Price**' as held against the Catalogue record

Custom Product: A Custom Product is always issued from stock (having previously been received from an Assembly Order). As a Custom Product is always Serial Numbered one of two costs will accompany the issued Item

- If the Custom Product Item record is flagged as '**Actual Costing is Used**' then, because the Custom Product is Serial Numbered, the cost will be the Actual Cost of the Assembly Order.
- If the Item record is not flagged as '**Actual Costing will be used**' then the Inventory Costing method will be used

2.2. Assembly Orders - Issues Costing

Assembly Order Issues can include

- Item Codes
- Descriptor Codes
- Labour Codes (for Planned Costs)
- Employee Bookings (for Actual Costs)

For each of these you would determine Planned and Actual Costs from the following sources.

Planned Costs

Items: When an Item is added to an Assembly Order the Costing Method used is that defined in the Assembly Rules (**Assembly>Settings>Assembly Rules**).

Descriptors: When a Descriptor is added to an Assembly Order the Costing Method used is that defined in the Assembly Rules (**Assembly>Settings>Assembly Rules**).

Labour Code: When a Labour Code is added to an Assembly Order the Planned Cost will always be the **Standard Cost** from the Labour Code record (**Labour>Labour Codes**).

Actual Costs

Items: An Assembly Order Line can have its 'Line Source' of

- Supply from Stock, or
- Source on Demand (linked to an Assembly or Purchase Order)

And this affects the Actual Cost that will be applied

Supply From Stock: The cost of the Item is determined by the 'Inventory Costing Method' as defined in **File>System Configuration>System Settings**. However,

- If the Item (**Inventory>Items**) is Serial Numbered controlled and is flagged as 'Actual Costing is Used' then the Actual Cost held against the issued Serial Numbered Item will be used.
- If the Item (**Inventory>Items**) is Batch Controlled and is flagged as 'Actual Costing is Used' then the 'Batch Average' Cost held against the Item's Batch Number will be used.

Source on Demand: The Item can be supplied directly from a linked Assembly Order or Purchase Order.

Purchase Order. The Actual Receipt Cost (as entered into **Purchasing>Purchase Receipts**) will be used.

Assembly Order. The Actual Receipt Cost (as defined in the section below (Assembly Order - Receipts) will be used.

Descriptors: A 'Descriptor' Assembly Order Line can be 'Sourced By'

- Internal, or
- Purchased (linked Purchase Order)

And this affects the Actual Cost that will be applied

Internal: The cost that will be applied will be related to the 'Descriptor Costing Method' as defined in **File>System Configuration>System Settings**.

The options are Last, Standard, or Buy Price

Purchased: The Descriptor can be supplied directly from a linked Purchase Order in which case the Actual Receipt Cost (as maintained in **Purchasing>Purchase Receipts**) will be used.

Labour: An 'Employee' must book time against 'Labour' Activities for the cost to be

applied. Go into the Employee record (**Labour>Employees**) and click on the '**Labour Codes**' tab. In that screen you will see that it contains an 'actual' cost value for each 'Labour Code'. When the Employee books time against a specific Labour Code the linked Cost as entered in this screen will be used. If the Employee record does not contain a cross reference to the Labour Code then the '**Unit Cost**' held against the Employee record will be used

2.3. Assembly Orders - Receipts Costing

In an Assembly Order the Receipt can include

- Main Product
- Co-Products
- Bi-Products

The program will first find out how you are going to Cost the Receipt. If you go to **File>System Configuration>System Settings** you will see field '**Inventory Costing Method**'.

If this is set to '**Standard**' then the Standard Cost as held against the Item, Co-Product, or By-Product will always be used irrespective of the percentage split in the BOM.

If this is set to '**Average**' then the 'Actual' Cost of the Receipt will be used. The Actual Cost of the receipt is determined by the setting in Assembly Rules'.

Go to **Assembly>Settings>Assembly Rules** and refer to field '**Default Receipt Cost Method**'. Two options are available

Planned Costs

The 'Planned Cost' of an Assembly Order is derived from the 'Planned' Costs of all the lines in the Assembly Order. This is divided by the Order quantity to arrive at a '**Unit Cost**'. The Planned Costs of each line Type is described above

Actual Costs

The '**Unit Cost**' is the 'Actual' cost of the components that have been issued at the time of the receipt divided by the Order Quantity.

The derived '**Unit Cost**' is further adjusted as follows:

Main Product: If the Assembly Order contains Co-Products then the Receipt Cost will be the above '**Unit Cost**' multiplied by

$$(100 - \text{sum of Co-Product percentages}) / 100$$

Co-Product: The Receipt Cost will be the above '**Unit Cost**' multiplied by the percentage contribution made by the Co-Product

By-Product: The Receipt Cost will be the above '**Unit Cost**' multiplied by the percentage contribution made by the By-Product

2.4. Job Order Costing

In a Job Order Lines you can include

- Item Codes
- Descriptor Codes
- Labour Codes
- Kitset Codes
- Catalogue Codes
- Custom Products (i.e. A product designed to a Customer's specific requirement)
- Task Bills

For each of these you would determine Planned and Actual Costs from the following sources.

Planned Costs

Item Codes: If you go into *Inventory>Items* then go to the 'Detail' tab then click on the 'Costing' Button you will be taken to the Item Costing screen. (This can also be accessed via *Pricing>Item Costing*). Each Item contains 3 costs (**Standard, Average, Last**). When an Item is added to a Job Order the Costing Method defined as the Planned Item Cost Method (as selected in *Job>Settings>Job Rules*) is used. This can be the Item's **Last Cost, Average Cost, Standard Cost, or Buy Price**. Note: The Buy Price is the price currently in the 'Std Buy Price' field on the Item's Detail screen

Descriptors: If you go into *Inventory>Descriptors* then go to the 'Detail' tab then click on the 'Costing' Button you will be taken to the Descriptor Costing screen. (This can also be accessed via *Pricing>Descriptor Costing*). Each Descriptor contains 2 types of cost (**Standard and Last**).

When a Descriptor is added to a Job Order the Costing Method defined as the Planned Item Cost Method (as selected in *Job>Settings>Job Rules*) is used. This can be the Descriptor's **Last Cost, Standard Cost, or Buy Price**. Note: The Buy Price is the price currently in the 'Std Buy Price' field on the Descriptor's Detail screen

Labour Code: If you go into *Labour>Labour Codes* and then go to the 'Detail' tab you will see that the Labour Code contains a 'Standard Cost'. You can manually enter the Cost here or, if you click on the 'Standard Cost Breakdown' button to the right of that field, you will see that you can (optionally) define that Standard Cost as being comprised of **Direct Labour, Fixed and Variable Overheads**. When a Labour Code is added to a Job Order the Costing Method defined as the Planned Item Cost Method (as selected in *Job>Settings>Job Rules*) is used. This can be the Labour Code's **Last Cost, or Standard Cost**.

Kitset: A Kitset is an accumulation of components (Items and Descriptors). When a Kitset is added to a Job Order the Costing Method defined as the 'Planned Kitset Cost Method' (as selected in *Job>Settings>Job Rules*) is used. This can be the Kitset Header's (I.e. Descriptor's) **Last Cost, Standard Cost, Buy Price**, or it can be **Calculated** (Derived from the sum of the costs of the Kitset Components. These Components use the Costing Methods given above)

Catalogue Code: The Planned issue Cost of Catalogue Code items is the 'Buy Price' as held against the Catalogue record

Custom Product: The Cost of Custom Product is derived from functions within the Custom Product script. If the costing function in the script is not used then the Item/Descriptor/Catalogue Code costs described above will be used

Task Bill: A Task Bill is an accumulation of components (Items, Descriptors, and Labour Codes). When a Task Bill is added to a Job Order the Costing Method defined as the Planned Task Bill Cost Method (as selected in *Job>Settings>Job Rules*) is used. This can be the Task Bill Header's (I.e. Descriptor's) **Last Cost, Standard Cost, Buy Price**, or it can be **Calculated** (Derived from the sum of the costs of the Task Bill Components. These Components use the Costing Methods given above)

Actual Costs

Items: A Job Order Line can have its 'Line Source' of

- **Supply from Stock**, or
- **Source on Demand** (linked to an Assembly or Purchase Order)

And this affects the Actual Cost that will be applied

Supply From Stock: the cost that will apply will be in the following order of priority:

- If the Item (*Inventory>Items*) is Serial Numbered controlled and is flagged as '**Actual Costing is Used**' then the Actual Cost held against the issued Serial Numbered Item will be used.
- If the Item (*Inventory>Items*) is Batch Controlled and is flagged as '**Actual Costing is Used**' then the 'Batch Average' Cost held against the Item's Batch Number will be used.
- The '**Inventory Costing Method**' as defined in *File>System Configuration>System Settings* will be used. The options are **Average** or **Standard**

Source on Demand: The Item can be supplied directly from a linked Purchase or Assembly Order.

Purchase Order. The Actual Receipt Cost (as entered into *Purchasing>Purchase Receipts* will be used.

Assembly Order. The Actual Receipt Cost (as defined in section above (Assembly Order - Receipts0 will be used.

Descriptors: A 'Descriptor' Job Order Line can be '**Sourced By**'

- Internal, or
- Purchased (linked Purchase Order)

And this affects the Actual Cost that will be applied

Internal: The cost that will be applied will be related to the '**Descriptor Costing Method**' as defined in *File>System Configuration>System Settings*.

The options are Last, Standard, or Buy Price

Purchased: The Descriptor can be supplied directly from a linked Purchase Order in which case the Actual Receipt Cost (as maintained in *Purchasing>Purchase Receipts* will be used.

Labour: An 'Employee' must book time against 'Labour' Activities for the cost to be applied. Go into the Employee record (*Labour>Employees*) and click on the '**Labour Codes**' tab. In that screen you will see that it contains an 'actual' cost value for each 'Labour Code'. When the Employee books time against a specific Labour Code the linked Cost as entered in this screen will be used. If the Employee record does not contain a cross reference to the Labour Code then the '**Unit Cost**' held against the Employee record will be used

Kitset: The cost of the Kitset is derived from sum of the individual costs of the components in the Kitset that were actually issued to the Job Order. As the components can be Items or Descriptors then the above 'Actual' Cost methods will be used

Catalogue Code: The Actual Receipt Cost (as entered into *Purchasing>Purchase Receipts*) will be used.

Task Bill: The Actual cost of the Task Bill is derived from the individual costs of the Task Bill components that were actually issued to the Job Order. As the components can be Items, Descriptors, or Labour Codes then the 'Actual' Cost methods described above will be used.

Custom Product: A Custom Product is always issued from stock (having previously been received from an Assembly Order). Because it is always Serial Numbered one of two costs will accompany the issued Item

- If the Custom Product Item record is flagged as '**Actual Costing is Used**' then, because the Custom Product is Serial Numbered, the cost will be the Actual Cost of the Assembly Order
- If the Item record is not flagged as '**Actual Costing will be used**' then the Inventory Costing method will be used

2.5. Purchase Order Costs

In a Purchase Order Lines you can include

- Item Codes
- Descriptor Codes
- Catalogue Codes

Ostendo does not address the anticipated Costs but simply focuses on the Buy Price as the basis for creating a Purchase Order. The Actual Costs are only known at the Receipt/Invoice time where additional costs such as Duty, Packaging, Transportation, etc can be identified.

Buy Prices

Items: A Buy Price matrix can be set up that has the following:

- A Base Buy Price
- A Supplier Price with quantity discounts

Descriptors: A Buy Price matrix can be set up that has the following:

- A Base Buy Price
- A Supplier Price with quantity discounts

Catalogue Item: A Buy Price from the Supplier with quantity discounts applied

Actual Costs

Order Receipt: During Purchase Order receipt the Receipt Cost is prefilled with the **Buy Price**. Option is provided to amend this if required. Additional costs can be added relative to the **Shipment Costs** that accompanied the Receipt. This can be apportioned across all lines in a Receipt or allocated to specific lines. This results in each Receipt Line having an accurate 'Actual Cost', which is used as follows:

Item Codes: Two scenarios can arise here:

- If the Item is NOT linked to a Job Order then it will be received into Stock. When receiving into stock the following options are used:
 - If the Inventory Costing Method is '**Average**' then the receipt Cost will be applied to the Receipt.
 - If the Inventory Costing Method is '**Standard**' then the Standard Cost will be applied to the Receipt and any difference posted to a Receipt Variance.
- If the Item is linked to a Job Order then the receipt Cost will be posted to the Job. However:
 - If the Job is NOT '**Closed**' then the cost will be posted to the Job Order Line
 - If the Job is '**Closed**' then the Job will be re-opened and the cost posted to the Job Order Line

Descriptors: Two scenarios can arise here:

- If the Descriptor is NOT linked to a Job Order then the cost will be posted to the Cost Centre held against the Purchase order Line ('Line Allocations' tab). Note: This Cost Centre will be:
 - The Cost Centre held against the Descriptor Record or,
 - Cost Centre '**DESCRIPTOR EXPENSES**'.
- If the Descriptor is linked to a Job Order then the receipt Cost will be posted to the Job. However:
 - If the Job is NOT '**Closed**' then the cost will be posted to the Job Order Line
 - If the Job is '**Closed**' then the Job will be re-opened and the cost posted to the Job Order Line

Catalogue Code: Two scenarios can arise here:

- If the Catalogue Code is NOT linked to a Job Order then the cost will be posted to the Cost Centre held against the Purchase order Line ('Line Allocations' tab).
Note: This Cost Centre will be:
 - The Cost Centre held against the Catalogue's Header Record or, if one has not been defined against the Descriptor record
 - Cost Centre '**CATALOGUE EXPENSES**'.
- If the Catalogue Code is linked to a Job Order then the receipt Cost will be posted to the Job. However:
 - If the Job is NOT '**Closed**' then the cost will be posted to the Job Order Line
 - If the Job is '**Closed**' then the Job will be re-opened and the cost posted to the Job Order Line

Supplier Invoice Receipt: If the Invoice Value is at variance to the Receipt Cost then the following actions will be carried out.

Item Codes: Two scenarios can arise here:

- If the Item is NOT linked to a Job Order then the difference between the Receipt Cost and Invoice Value will be posted to a Purchase Price Variance Account
- If the Item is linked to a Job Order then:
 - If the Job is NOT '**Closed**' then the difference will be posted to the Job Order Line
 - If the Job is '**Closed**' then the Job will be re-opened and the difference posted to the Job Order Line

Descriptors: Two scenarios can arise here:

- If the Descriptor is NOT linked to a Job Order then the difference between the Receipt Cost and the Invoice Cost will be posted to the Cost Centre held against the Purchase order Line ('Line Allocations' tab). Note: This Cost Centre will be:
 - The Cost Centre held against the Descriptor Record or, if one has not been defined against the Descriptor record
 - Cost Centre '**DESCRIPTOR EXPENSES**'.
- If the Descriptor is linked to a Job Order then:
 - If the Job is NOT '**Closed**' then the difference between the Receipt Cost and the Invoice Cost will be posted to the Job Order Line
 - If the Job is '**Closed**' then the Job will be re-opened and the difference posted to the Job Order Line

Catalogue Code: Two scenarios can arise here:

- If the Catalogue Code is NOT linked to a Job Order then the difference between the Receipt Cost and the Invoice Cost will be posted to the Cost Centre held against the Purchase order Line ('Line Allocations' tab). Note: This Cost Centre will be:
 - The Cost Centre held against the Catalogue's Header Record or, if one has not been defined against the Descriptor record
 - Cost Centre '**CATALOGUE EXPENSES**'.
 - If the Catalogue Item is linked to a Job Order then:
 - If the Job is NOT '**Closed**' then the difference between the Receipt Cost and the Invoice Cost will be posted to the Job Order Line
 - If the Job is '**Closed**' then the Job will be re-opened and the difference posted to the Job Order Line
-

3. Inventory Costing

Three cost types are always maintained against all Items. These are:

- **Standard Cost:** As held against the Item Master's Cost record
- **Average Cost:** Adjusted by Stock Movements (see below)
- **Last Cost:** Adjusted by last Stock Receipt transaction

Additionally a fourth Cost Type is maintained against selected Items

- **Actual Cost:** Issues and Receipts at Batch/Serial Number level

3.1. Standard Cost

Whenever a stock movement is carried out (Issue or Receipt) then the Item's Standard Cost will be applied to the Item.

3.2. Average Cost

Whenever an Item is Received then the Average Cost is adjusted using the following formula

$$\text{Item Average Cost} = \frac{(\text{Current Ave} * \text{Current Qty}) + (\text{Received Qty} * \text{Received Cost})}{(\text{Current Qty} + \text{Received Qty})}$$

Whenever an Item is issued then the Item's Average Cost is used

However, the following should also be noted.

- If an Item is being issued that forces the current stock into negative then the current Average Cost will still apply
- If a Receipt is being made and the current stock is negative then the following costs will be applied.
 - If the Receipt Quantity results in the finished stock balance remaining negative then the current Average Cost will be applied to the full receipt quantity. However, the difference between the Receipt's Actual Cost and the Inventory Average Cost will be posted to a Stock variance.
 - If the Receipt Quantity results in the finished stock balance being positive then the quantity to bring the current stock to zero will be received at the current Average and the balance received at the receipts Actual. However, the difference between the Receipt's Actual Cost and the evaluated Receipt Cost will be posted to a Stock variance.
- Where a (new) Item is being issued but no prior receipt transaction has been made and the Average Cost is zero then a message will be returned to warn that the Issue Cost will be zero unless action is taken to update the Average Cost field. This can be done manually against the Item Master record. Subsequent Inventory Cost adjustments will be made as outlined in (b) above

3.3. Last Cost

Whenever a stock receipt is carried out then the Last Cost field against the Item will be updated.

3.4. Actual Cost

A pre-condition of Actual Cost is that the Item must be Batch or Serial Controlled. Whenever a stock receipt is carried out then a record of the Batch/Serial Number is created and the Actual Cost is attached to this record. Upon Issuing the Item a Batch Number/Serial Number is required

and hence the Actual Cost is carried along with the issue

Note: A Batch can contain more than one receipt, each having different Actual Cost. In this instance a 'Batch Average' will be used.

3.5. Cost Revision History

All changes to Standard, Average, and Last Cost will be maintained within Ostendo giving a complete history of Type of Change (Standard, Average, Last) along with date of change plus before and after costs.

4. BOM Costing

Bills Of Material are used in Assembly Jobs to make standard products for stock. BOM Costing allows the user to carry out a multi-level Cost Roll-up using the Standard Cost for Items, Descriptors and Labour plus Fixed and Variable Overheads related to Labour.

The Cost Roll up program goes through the following steps:-

- Sort all parent Items in the BOM table into Low-Level Code descending sequence
- Commencing with the first parent calculate the following from its components

Set Up Cost

Items	- Standard Cost * Usage Quantity / Setup Batch Size
Descriptors	- Standard Cost * Usage Quantity / Setup Batch Size
Labour	- Labour Code * Labour Cost * Hours / Setup Batch Size
FOH	- Labour Code * FOH Cost * Hours / Setup Batch Size
VOH	- Above Labour calculation * VOH recovery Rate

Process Cost

Items	- Standard Cost * Usage Quantity
Descriptors	- Standard Cost * Usage Quantity
Labour	- Labour Code * Labour Cost * Hours
FOH	- Labour Code * FOH Cost * Hours
VOH	- Above Labour calculation * VOH recovery Rate

- Progressively repeat for all parent Items in the BOM
- Select Items whose Standard Cost is to be updated

Note: The Labour Cost (Labour, FOH, & VOH) for an activity comes from a Labour Code rather than a specific Employee.

If the BOM contains Co-Products then the percentage allocated to the Co-Products is deducted from the Costs of the BOM lines to arrive at the BOM's Parent Item Cost

5. Employee Costing

A Labour Code is used in BOMs, Jobs Orders, Assembly Orders, Custom Product Orders, etc. to derive a Planned Cost of the Labour activities. The Actual Labour Cost comes from the Employee who carried out the task. Let's look at this in more detail.

Each Employee record has a cross-relationship whereby when the Employee carries out work against a Planned Labour Code then a defined Actual Cost rate is applied. For example let us assume that two Labour Codes have been created:

Labour Code LAB001 uses Std Cost \$20.00, Fixed OH of \$25.00 and Variable OH of 150%
 Labour Code LAB002 uses Std Cost \$22.00, Fixed OH of \$33.00 and Variable OH of 140%

These Costs are used for Planned Costs during BOM Roll-up, etc.

Employee EMP011 could have the following information held against it.

If working in LAB0001 then Std Cost \$24.00, Fixed OH of \$28.00 and Variable OH of 100%

If working in LAB0002 then Std Cost \$26.00, Fixed OH of \$28.00 and Variable OH of 150%

Therefore if Employee EMP011 booked time against an activity planned for Labour Code LAB001 then each hour booked would incur an Actual Cost of

Labour Charge	\$24
Fixed Overhead	\$28
Variable Overhead	\$24

However, if Employee EMP011 booked time against an activity planned for Labour Code LAB002 then each hour booked would incur an Actual Cost of

Labour Charge	\$26
Fixed Overhead	\$28
Variable Overhead	\$39

6 5. Customers Sales Orders, Rentals and Invoicing

This section will include the creation and maintenance of Customer information then continuing through the various Order processes

6.1 Customers

Customers are referenced across Ostendo in all of the operational modules (CRM, Sales Orders, Job Orders, Service Orders, Inventory, Pricing, Purchasing, etc)

1. Preparation

The following tables are used when creating Customer records. Take a look at them. There are some defaults already set up but you may wish to add more or amend the current records:

Mandatory Tables

The following three fields are mandatory and validated against separate tables when creating a Customer record. Within each Table, however, you can nominate a 'default' that will populate a Customer record when adding a new Customer record.

Customer Types: Segregates Customers into logical groups (e.g. Trade, Retail). You can maintain these via [Sales>Settings>Customer Types](#)

Tax Group: To facilitate Customer/Item Tax code evaluation. You can maintain these via [File>Financial Configuration>Credit Terms](#)

Terms: Days from (Invoice, EOM, End of Next Month). You can maintain these via [File>Financial Configuration>Tax Groups](#)

Optional Tables

The following fields are optional and, when used, are validated against a separate table when creating or maintaining a Customer record.

Sales Regions: For Sales Analysis purposes. You can maintain Sales Regions via [Sales>Settings>Sales Regions](#)

Customer Codes: A Customer Code can be used to 'group' Customers within the same group (Example: Retail, Trade, etc). To maintain Customer Codes go into [Sales>Settings>Customer Codes](#)

Shipping Methods: This allows you to pre-define that Shipping Method applicable to the Customer and if the method is Taxable (along with the Tax Rate). To maintain Shipping Methods go into [Sales>Setting>Customer Shipping Methods](#)

Sales Person: You can allocate a default Salesperson to a Customer. This can be amended at Sales Order level if required. To create a Salesperson go into [Labour>Employees](#) and create an Employee. In the main Employee screen you can identify the Employee as being a Salesperson

Lead Source: This can be used for Sales Analysis purposes. To maintain Lead Sources go into [Sales>Settings>Customer Lead Sources](#)

Price Level: This is used in the Pricing function to determine the sell price of an Item or Descriptor based upon the Pricing Level held against this Customer. Examples of Pricing

Levels are Retail, Trade, etc. To maintain Price Levels go into **Pricing>Settings>Price Levels**. The use of Price Levels will be covered in more detail later in this document

Rate Level: This is used in the Pricing function to determine the Charge Rate for a Labour activity based upon the Pricing Level held against this Customer. Examples of Rate Levels are Retail, Trade, etc. To maintain Rate Levels go into **Pricing>Settings>Rate Levels**. The use of Rate Levels will be covered in more detail later in this document

Invoicing Group: This is (optionally) used when generating and printing Invoices where you can restrict the batch selection to Customers within a specific Invoicing Group. To maintain Invoicing Groups go to **Sales>Settings>Invoicing Groups**.

Statement Cycle Code: Enables Statements to be printed relative to their 'cycle'. To maintain Statement Cycle Codes go to **Sales>Settings>Statement Cycles**

2. Create Customer records

Go into **Sales>Customers** and click on the **'Add'** button. Add your own Customer details. You will see that the created record is prefilled with the defaults identified in the previous section. These fields can be amended if required.

Having created the Customer record the following linked records can be created and maintained

2.1. Customer Additional Fields

There are two levels where Additional fields could be required against Customers: Global Fields that apply to ALL Customers (Example:- 'Valued Customer' flag, etc) and Properties that apply to some Customers (Example:- Overseas 'Continent')

2.1.1. Global Additional Fields

Let us create an additional field called **'Valued Customer'** linked to a Yes/No checkbox

Go into **File>System Configuration>Additional Fields** and click on the **'Add'** button. On the displayed line enter the following:

Module: Select **'Customers'**

Caption: Enter the Additional Field name (In our example: **Valued Customer**)

Field Type: From the drop-down list select the format of the field. The options are:

- **Text:** Any data format can be entered in a Text field
- **Decimal:** Allows entry of numbers and decimals
- **Integer:** Allows entry of whole numbers only
- **Currency:** Shows Currency symbol and decimals as defined in Regional Settings
- **Yes/No:** Shows a checkbox which can be checked/unchecked
- **Date:** Contains a drop-down calendar for selection of a date
- **Time:** Displays format HH:MM:SS for entry of a time of day

Value List: This allows you to define any specific entries to which a drop-down list - during data entry - is restricted. Leave this blank

'Save' the entry and **'Close'** the screen when done

If you now go to the Customer screen (**Sales>Customers**) and click on the **'Detail'** tab you will see an **'Additional Fields'** tab in the centre-left of the screen. Click on this tab and 'check' the checkbox to denote that this is a valued Customer then **'Save'** the record.

You can view these additional fields in the Customer **'List'** view if required by going into the

Customer's 'List' screen and 'right mouse' in the centre panel. Select 'Customize List Fields' from the displayed panel. (Note: If that option is not visible then go to **File>System Configuration>User Security and Options** and go to the 'User Options' tab for the current User. 'Check' both the 'Save Grid Layouts' and 'List Customising' checkboxes.) On the displayed panel 'check' 'Additional Field_1' and give it a 'Display Name' of (say) 'Valued Customer'. 'Save' the entry. The field will now display on the List screen where you can sort and filter as necessary.

2.1.2. Customer Properties

This feature allows you to define a 'Property' (Example: **Corporate Status**) and then link that property to selected Customers with a value that is specific to each Customer (Example: Gold, Silver, Bronze)

To demonstrate this, go into **General>Settings>General Properties** and 'Add' 'Corporate Status' with Property Type of 'Text' and the following entries - on separate lines - in the Property Values field (**Gold, Silver, and Bronze**). Click on 'Save' and then 'Close'

Now go to **Sales>Customers** and select the customer you created above. Click on the 'Related' Button on the right of the screen and select 'Customer Properties'. Click the 'Add' button and:

- Add a line and, in the field 'Property' select 'Corporate Status' from the drop-down
- Select 'Gold' from the drop-down under column 'Value'
- 'Check' the 'Copy to SO Lines' checkbox

The selected property and value will accompany the Customer whenever a Sales Order is raised for this Customer. To demonstrate this go into **Sales>Sales Orders** and click the 'Add' button. Select the Customer you created above then click the 'Create Order' button. Click on the 'Related' button in the generated Order and select 'Sales Order Properties'. You will see that the Customer's Properties have been copied to the Sales order.

2.2. Customer Contacts

You can have multiple Contacts against each Customer. To create these contacts go to the Customer screen (**Sales>Customers**) and click on the 'Related' button down the right of the screen and select 'Contacts'. Alternatively you can go to **CRM>Contacts** and create the contacts via that screen using 'Contact Type' = 'Customer' and selecting this Customer under 'Company Name'.

2.3. Delivery Addresses

You can have multiple Delivery Addresses against a Customer. Any one of these can be selected during Order entry and made specific to the Order. To create additional Addresses click on the 'Additional Physical Addresses' button in the main Customer screen

2.4. Customer Images

You can add multiple images (pictures, drawings, maps, plans, etc) to a Customer record. These can be printed on all documents where the Customer is used. Go to **Sales>Customers** and select the Customer you created above. Click on the 'Related' Button on the right of the screen and select 'Customer Images'. Click the 'Add/Edit' button and:

- Give the Image a short Name
- Point the program to where the image is located on your computer network
- 'Check' the 'Copy to Sales' checkbox then save and exit the screen

2.5. Customer Documents

You can add multiple documents to a Customer record. These can be printed along with all documents where the Customer is used. Go to **Sales>Customers** and select the Customer that you created above. Click on the **'Related'** Button on the right of the screen and select **'Customer Documents'**. Click the **'Add/Edit'** button and:

- Give the Document a short Name
- Point the program to where the document is located on your computer network
- 'Check' the **'Copy to Sales'** box then save and exit the screen

2.6. History Notes

This function allows you to link multiple time-stamped notes to a Customer record. Against selected History Notes you can also add a dated reminder so that Ostendo will prompt you of the reminder once the date is reached. Go to **Sales>Customers** and select the Customer you created above. Click on the **'Related'** Button on the right of the screen and select **'Customer History Notes'**. Click the **'Add'** button and:

- Enter some history notes
- 'check' the **'Follow-Up'** required checkbox and select a date from the adjacent drop-down calendar. **'Save'** the History Note and exit the screen
- To see the **'Follow Up'** in action you should first change the company by clicking on **File>Change Company** and selecting **DEMO** then sign in as **ADMIN/pass**. Repeat this and go back to company **'Training'**. Upon sign-in as **ADMIN/pass** the alert should present itself if the Follow-Up date is current
- Note: In the CRM Module if a **'Call'** was raised and subsequently **'Closed'** against a Customer then any activity notes entered against that Call will be posted to the Customer's History file

6.2 The Sales Order Process

There are 2 Sales Order Styles used in Ostendo

- Raise Order > Pick Goods > Invoice
- Raise Order > Pick Goods > Despatch > Invoice

You can see that one has a Despatch function where the other doesn't. These variances will be covered in more detail later. In this section we will go through what is required to initially raise a Sales Quotation or Order.

1. Preparation

You may wish to address the following areas before creating a Sales Quote or Order

Auto Numbering: You can nominate numbers from which the program auto generates an incremental number for Sales Order identity. Go into **File>System Configuration>System Settings** then click on the **'System Numbering'** tab. All the references under the 'Sales' Module will be addressed in this exercise

Sales Types: This allows the creation of multiple Order Types. Each Order Type contains:-

Opening Status: The Order Status allocated to the Order when it is created

Numbering: Defines if it uses Auto Numbering or the Number is manually entered

Prefix: A 5-character prefix to the Order Number (For Example SO-, ORD, etc)

Style: Order is a 'Delivery' Order or a 'Counter' Order

Rental Order: This will be covered in Section 6 so leave this blank for now

To view current Sales Types or create your own go into **Sales>Settings>Sales Types**

Sales Rules: This allows you to define rules that apply to Sales Orders. Go into

Sales>Settings>Sales Rules and adjust the rules to suit your requirements

Invoice and Credit Name: Many countries have a specific name that must appear on all Invoices (Example, Tax Invoice, Invoice for an Invoice and Credit Note, Credit for a Credit, etc). The entry made in **File>System Configuration>System Settings** will appear as the main title to the printed Invoice or Credit Note.

2. The Quotation and Order Process

2.1. Create the Order

Go into **Sales>Sales Orders** and click the **'Add'** button. A panel will appear. You will see that the Order can be created 'from scratch', or by copying an existing Sales Order. Select Sales Type '**Counter Sales**'. From the drop-down list under **'Customer'** select the Customer you created above then click the **'Create Order'** button. We will look at the specifics relating to a Quotation in the next section.

Look at the fields in the Order Header and refer to Ostendo Help for more information

Click on the **'Lines'** tab to enter Sales Order Lines

2.2. Add Sales Order Lines

Many options are available for adding lines to the Sales Order such as:

- Selecting from a user-defined List
- Selecting Items in batch
- Selecting Descriptors in batch
- Selecting multiple Lines from a Supplier Catalogue
- Selecting a single Item
- Selecting a single Descriptor
- Selecting a single Line from a Supplier Catalogue
- Selecting a complete Kitset of Items/Descriptors
- Configuring a 'Custom Product'
- Quick Line entry

2.2.1. Selecting from a user-defined List

Click on the **'List'** button on the **'Batch Entry'** bar. A panel is presented that displays all pre-defined Lists. You will see that 2 Lists already exist in the database. Upon selecting a **'List'** a further panel will appear showing all lines in the List (Items, Descriptors, Labour). Select a couple of lines from the List and click the **'Create Lines from Selected Contents'** button. Each selected line will become a Sales Order Line in its own right.

There are two ways that you can present the List in the Sales Order process. This is defined against the List. Therefore go into **Inventory>Lists** and highlight **'Fastener-12mm-List'**. Go into the **'Detail'** tab and you will see field **'List Style'**. This is currently set to **'Check List'**. As you are already aware this style presents a list where your selection is 'checked' if you are selecting it and you can amend the **'Per Qty'**.

If you now amend the field **'List Style'** to read **'Order Pad'** we will see the second way that Lists are presented. Go back to the Sales Order Line screen and click on the **'List'** button on the **'Batch Entry'** bar. The presented panel is a straight Order Pad into which you can simply enter the quantity required (then press the **'Return'** or **'Down Arrow'** key on the keyboard to go to the next line). Any line with zero quantity will not be copied to the Sales order.

2.2.2. Selecting Items in batch

Click on the **Items** button on the **Batch Entry** bar. A panel is presented that displays all Items (excluding those with status **Obsolete**) in Ostendo. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the **Add to Selected** button

In the lower panel you can now amend the required quantity.

Once the full Item selection has been made then click the **Create Lines from selected contents** button to add the lines to the Sales Order

2.2.3. Selecting Descriptors in batch

Click on the **Descriptors** button on the **Batch Entry** bar. A panel is presented that displays all **Active** Descriptors that are designated 'for general purpose use'. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the **Add to Selected** button

In the lower panel you can now amend the required quantity.

Once the full Descriptor selection has been made then click the **Create Lines from selected contents** button to add the lines to the Sales Order

2.2.4. Selecting multiple lines from a Supplier Catalogue

Click on the **Catalogue Items** button on the **Batch Entry** bar. A panel is presented that shows all Supplier Catalogues; select the Catalogue from which Items are to be extracted. (Note: If you have only one Supplier Catalogue then this step is ignored by Ostendo).

All Items in the selected Supplier Catalogue are now displayed. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the **Add to Selected** button

In the lower panel you can now amend the required quantity.

Once the full selection has been made from the Catalogue then click the **Create Lines from selected contents** button to add the lines to the Sales Order

2.2.5. Selecting a single Item

This can be used as an alternative to the above 'Batch' selection. Click on the **Add** button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against **Line Type** select **Item Code** then go to the next field (**Code**) to select the specific Item from the drop-down list. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select any Item.

2.2.6. Selecting a single Descriptor

This can be used as an alternative to the above 'Batch' selection. Click on the **Add** button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against **Line Type** select **Descriptor Code** then go to the next field (**Code**) to select the specific Descriptor from the drop-down list. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select any Descriptor

2.2.7. Selecting a single Line from a Supplier Catalogue

This can be used as an alternative to the above 'Batch' selection. Click on the 'Add' button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against 'Line Type' select 'Catalogue Code' then go to the next field ('Code') to select the Catalogue from which Items are to be extracted. (Note: If you have only one Supplier Catalogue then this step is ignored by Ostendo). Having selected the Catalogue then all lines within that catalogue are displayed. Select the specific Catalogue Item that you want to add to the Sales Order Line. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select Item **PT-3220** from the 'Power Tools' Catalogue

2.2.8. Selecting a Serial Kit

Serial Kits are individual – unique - Serial Numbers within an Item. Each Serial Kit can comprise of one or more Items that may vary in their individual attributes (Colour, Size, Batch, Serial Number, Expiry Date, etc). When a Serial Kit Item is received into stock, and given a Serial Number, Ostendo will attach a 'Template' Kit 'Content' List as defined against the Serial Kit Item record. That kit can be 'personalised' to suit the Serial Kit Content. When adding a Serial Kit Item it is treated as any other Item. The difference occurs when it is 'Picked' in that you are required to select a specific Serial Numbered Kit. (A detailed tutorial of this function is given in Exercise 2 - Items Descriptors and Labour Codes).

2.2.9. Selecting from a complete Kitset of Items/Descriptors

A Kitset is a pre-defined 'kit' of Items and/or Descriptors that are sold as a single combined unit. To look at current Kitsets in Ostendo go to **Sales>Kitsets** where you will see a single Kitset 'SPARESKIT'. If you wish to create your own Kitset then carry out the following:

- Create a Descriptor (**Inventory>Descriptors**) and give it a Descriptor Classification (I.e. 'Check' the Radio Button) against 'The Descriptor is used as a Kitset Code'.
- Go into **Sales>Kitsets** and 'Add' a new Kitset by selecting the above Descriptor
- Go into the 'Lines' tab for the Kitset and add a couple of Items and Descriptors to the Kitset

Having created the Kitset (or using Kitset 'SPARESKIT') add it to the Sales Order Line by going to the Sales Order Lines screen and clicking the 'Add' button located to the right of the screen. This will allow you to add details of a single Kitset to be made in the lower part of the Order Line screen. From the drop-down against 'Line Type' select 'Kitset Code' then go to the next field ('Code') to select the Kitset from current list of Kitsets. All the remaining information (except the 'greyed out' fields) can be amended as required.

2.2.10. Configuring a Custom Product

Ostendo has two versions of Custom Products.

Manual: A simple selection of components and quantities from a pre-defined Bill of Material

Rules Based: A more complex Custom Product configurator with inbuilt conditional inclusions and exclusions, calculations, and other user-specific Rules

Exercises covering these can be found in Section 10 of Training Guide 5 - Assembly Orders

2.2.11. Quick Line Entry

Facility is available for quick entry of lines using the Item or Descriptors Barcode. To prepare for Speed Entry:

- Go into **Sales>Settings>Sales Rules** and 'check' the checkbox against field **'Order Speed Entry'**.
- Next step is to go into **Inventory>Items** and add a barcode to a couple of items
- Do the same in **Inventory>Descriptors** against a couple of Descriptors

Create a new Sales Order and click on the **'Lines'** tab. If you click the **F5** key than a new line will be generated. Manually enter one of the above barcodes into field **'Code'** (This emulates a barcode scanner) and the Item Number or Descriptor will populate this field and the cursor will go to the Qty field. Enter a quantity and click the **F5** key to go to the next line.

2.3. Additional Information in Sales Order Line

Dependent upon the type of Sales Order Line the following additional information is available

2.3.1. Line Info Band

Just above the lower detail panel you will see a band called **'Line Info'**. The following buttons are presented in this band where applicable:

Add-On Sales: If the Item has **'Add-On Sales'** then this button will allow you to ask any 'Add-On Sales' questions and take orders for those 'Add-On' Items being purchased. To see this in action we will (say) offer an Extended Warranty with the sale of an Item.

Go into **Inventory>Descriptors** and add a new Descriptor **'Extended Warranty'**. Within that screen:

- Enter a Sell Price of **\$20**
- 'Check' the **'Sales Warranty Applies'** checkbox
- Select **36M-PANDL** from the drop-down list in the adjacent field

Go into **Inventory>Items** and select **WAGON-2189**. 'Check' the **'Add On Sales Apply'** checkbox and click on the adjacent **'Item Add-On Sales'** button. In the presented panel add the **'Extended Warranty'** Descriptor.

Finally, using the Sales Order created above, add **'WAGON-2189'** to the line. The **'Add-On Sales Items'** button will be displayed. If you click on this button then you can offer the Add-On Sales Items for inclusion with this Order Line

Qty Break Pricing: If an Item or Descriptor has Quantity Break Pricing then this button will be presented which, when pressed, will show the Quantity Break details. To see this in action:

- Go into the Customer record that you created in Exercise 2. On the Customer 'Detail' screen click on the **'Pricing and Invoicing'** tab then select **'Retail'** from the drop-down list under **'Pricing Level'**.
- Go into **Pricing>Item Pricing** and select **'WAGON-2189'** then click on the **'Detail'** tab. Add a new line with Price Level **'Retail'** and apply your own Price Breaks
- Finally, if you haven't already done so, create a 'Counter' type Sales Order for the above Customer and add **'WAGON-2189'** to the line. The Qty Beak Pricing button will now appear on the 'Line Info' band

Alternate Items: If an Item has an Alternative Item referenced to it then this button is presented so that you can view information about that alternative. To see how this works let us suppose that Item **5000-2011** (Cat 6 Network Cable - 5 Metres) could be supplied as an alternative to Item **5000-2010** (Cat 6 Network Cable - 1.2 Metres). Go into **Inventory>Items** and select Item **5000-2010**. In the Detail view click on the **'Additional Inventory Settings'** button and 'check' the **'Alternate Item Available'** checkbox. Select Item **5000-2011** from the drop-down list in the field immediately underneath the checkbox.

Finally, using the Sales Order created above, add **'5000-2010'** to the line. The Alternate Item button will now appear on the 'Line Info' band.

Stock: If any Order Line is an Ostendo Item then this button will be displayed. It shows the current On-Hand quantity (in stock), Available Quantity (On-Hand + Supply - Demand) plus the Item's basic Unit of Measure. Clicking on this button will show further details.

2.4. Additional 'Tabs' applicable to the Sales Order Line

Dependent upon the type of Sales Order Line the following 'Tabs' will be presented

Line Source: Based upon the 'Line Type/Supply Method/Sourced By' combination this panel shows from where the line is provisioned. The following options are covered.

- Item Code/Internal - An Assembly Order will be raised to cover this demand
- Item Code/Purchasing - A Purchase Order will cover this demand
- Descriptor/Internal - No Supply Order planned but internal Resource is scheduled
- Descriptor/Purchasing - A Purchase Order will cover this demand
- Catalogue Item - A Purchase Order from the Catalogue Supplier will be raised

You should note that you can split the Line Source record to procure the line from multiple Sources. To do this click on the **'Add'** Button. This will create a new line into which you can add the new procurement source. You will also see a button **'Edit Source Quantities'** just above the created line. Clicking on this button brings up a separate panel for re-allocation of Order Line's quantities.

If you now select the Catalogue Item in the upper panel you will see that this line has a source linked to the Catalogue Supplier. If you click on the **'Related'** Button to the right of the screen then select **'Create Required Purchase Orders'** a screen will be presented for you to convert the Purchased **'Source On Demand'** Lines into physical Purchase orders. Try converting the above Catalogue Item ('Check' the Select checkbox and click the **'Generate Orders for selected requirements'** button). If you return to the **'Line Source'** tab you will see the Purchase Order reference appears in the last field.

Line Properties: Any specific property values held against an Item, Descriptor, Kitset, or Catalogue Item are copied to the Sales Order Line. You have the option to amend or delete current properties or even add new properties.

To see this in action you should first create the 'Property' via **General>Settings>General Properties** and add **'Voltage'** with Property Type of 'Text' and the following entries - on separate lines - in the Property Values field (**115 Volts**, and **230 Volts**). Click on **'Save'** and then **'Close'**

Now go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the **'Related'** Button on the right of the screen and select **'Item Properties'**. Click the **'Add'** button and:

- Add a line using the above property
- Select the Voltage from the drop-down list
- Select the specific Voltage from the drop-down under column 'Value'
- 'Check' the **'Copy to SO Lines'** checkbox

The selected property and value will accompany the Item whenever it is used in a Sales Order. To demonstrate this go into the Sales Order that you were using above and select the **'Lines'** tab and add an Order Line for the above Item. Click on the 'Line Properties' tab and the Properties will have been copied from the Item **485-2267**. You will see the 'Line Properties' appear on the **'Line Info'** bar

Planned Variants: This is visible if the line covers an Item that has 'sub-level' variations of Colour,

Size, or Grade. You can define the specific variants required against this Sales Order Line by clicking on the 'Add' button and entering the variants. The total quantity of sub-level variants must equal the Line Quantity. To see this in action you should go into **Inventory>Items** and select Item **485-2268** (Internal Spotlight 250 Watt). 'Check' the '**Colour**' checkbox in the '**Detail**' screen. Click on the '**Colours**' Button to the right of the checkbox and add colours '**Red**' and '**Green**'. Close out of the Item, screen and then go to the Sales Order that you were using above and select the '**Lines**' tab and add an Order Line for Item **485-2267**. Click on the 'Line Variants' tab and enter the quantity of each variant required in the Sales Order.

Picked Lines: This covered in Exercise 5.

Kit Contents: When a Kitset is copied to the Sales Order Line its content is shown here. The quantity per kit against each Line can be amended if required. The actual Issue of the Kit is carried out via the 'Picked Lines' tab. If you go into **Sales>Kitsets** you will see that a kitset has already been entered. Click on the 'Lines' tab to view the 3 entries that comprise the Kitset. Go to the Sales Order that you were using above and select the '**Lines**' tab and add an Order Line. In the drop-down under '**Line Type**' select '**Kitset Code**' then select the Kitset from the dropdown under '**Code**'. You will see that the tab '**Kitset Contents**' will now be displayed on the Line Info bar. Click on this to view the content of the Kit

Warranty: This enables you to adjust planned Warranty records that may exist against this Line or add new Warranty records as required.

To see this in action go into **Service>Warranty Definitions** and view the current Warranty Codes that are in Ostendo. Add your own as required. Now go to into **Inventory>Items** and select Item **1800-2190** (Rear Wheel Assembly). 'Check' the '**Sales Warranty Applies**' checkbox in the '**Detail**' screen. Click on the drop-down to the right of the checkbox and select one of the Warranty definitions then '**Save**' the record.

Go to the Sales Order that you were using above and select the '**Lines**' tab and add an Order Line and select Item **1800-2190**. If you click on the 'Warranty' tab you will see that the Warranty definition has been copied to the Sales Order Line. You can amend this, or add new Warranty records that will be specific to this Order Line.

Prices: This is for information only and shows past Sales with Quantities and Prices to this Customer.

2.5. Order Line Notes

At the bottom of the 'Lines' screen there is space to put unlimited Notes that apply to the Order Line. If you click in the Notes area you will see two Icons appearing in the top-right of the field. If you click on the first Icon then the Notes field will occupy a much larger area so that you can see the full content of your notes.

Before we go to the second Icon go to **General>Frequently Used Text** and create a common Text Message such as "This Item comes with a 36 Month Warranty covering Parts and Labour". Having done that go back to the Sales Order Line and click on the second Icon. A separate screen will appear showing the Frequently Used Text message that you have just created. If you 'double click' on the selected text, to highlight it and click the OK button then the text will be copied to these Notes.

6.3 Creating and Monitoring a Quotation

In the previous exercise you went through the various options to create a Sales Order. You should note that the creation of a Sales Quote includes all of the above processes to generate the Quote. These exercises describe the specific functions that support the generation and monitoring of the Quote

1. Quotation Styles

Within Ostendo you can use any one of 3 pre-defined Quote Styles.

- **Formal:** This shows all Line information along with each line's Unit Price, quantity and extended Price. All this is summated into a single Quote Price
- **Letter Style:** This is in the forma of a Letter that contains an introductory text (such as "Thank you for giving us the opportunity....etc) followed by a single line showing the Quoted Price and ending in a 'footer' Text (such as "This Quote is valid for a period of 30 days.....etc)
- **Letter Style plus Job Line information:** A combination of the above two Quote Styles

In preparation for generating the various styles go into **General>Frequently Used Text** and make two entries such as:

Name: Quote Header

Text/Phrase: Thank you for giving us the opportunity to make a Quotation. I'm sure that you will find our quoted price very competitive.

Name: Quote Footer

Text/Phrase: This Quotation is valid for a period of 30 days from the above date. If you have any questions please feel free to contact us at any time.

2. Quotation Settings

Ostendo has default settings for a Quotation that will be used to prefill the Quote Layout. You can, of course, amend make changes when printing the Quote. Go to **Sales>Settings>Sales Rules** where you will see the following fields that relate to a Quote

Quote Expiry Days: The number of days that the quote will remain valid. Currently this is set to **30** but you can amend this if required

Default Quote Header Notes: From the drop-down list select the '**Quote Header**' that you created above.

Default Quote Footer Notes: From the drop-down list select the '**Quote Footer**' that you created above.

Default Quote Style: Leave this as '**Formal**'

Default Quote Print Lines: From the drop-down list select the '**Print Single Line**' that you created above.

Default Quote Line Description: Enter a short text such as '**We are pleased to offer a quote of:**'

Save the changes

3. Creating the Quotation

Go into **Sales>Sale Orders** and click the 'Add' button. Select Customer '**Jim Gold & Co Ltd**' then click the '**Create Quote**' button. Go into the '**Lines**' tab and add two lines:

Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **50**

Line Type: **Descriptor Code**, Code: **MISCCHARGE**, and Order Qty: **150**

then return to the '**Detail**' view.

You will see the following fields that are specific to a Quotation.

- Quote Status:** The Status is 'Quote' and cannot be changed except by the program
- Quotation Print Status:** Field 'Quotation Print Status' has an adjacent button from which you can print the Quote
- Quote Expiry:** This is calculated from the System Date using the number of days set up in **Sale>Settings>Sales Rules**
- Quote Opt:** This allows you to adjust the content of the Quote.

Click on the **Quote Opt** button and a separate panel will be displayed. This shows the defaults that you set up above. Let us now look at the various Quote Styles that can be printed.

- **Formal:** If you click on the 'Print Quote' button then you will see the **Formal** style Quote.
- **Letter Style 1:** Select 'Letter' from the drop-down against 'Quote Style' then 'Save' the setting. If you now click on the 'Print Quote' button then you will see the **Letter** style Quote.
- **Letter Style 2:** You can see Style 2 by selecting 'Print Detail Lines' from the drop-down under field 'Lines to Print'. Now 'Save' the setting and click on the 'Print Quote' button. You will see that this is a combined 'Formal' and 'Letter' Quote
- **User:** The 'User' style is available for your Administrator to create a radically different Style without affecting the 'Formal' or 'Letter' Quote Styles. Currently this contains a copy of the 'Formal' quote style.

4. Line Grouping

There may be instances where, in a Quotation, you wish to group together similar lines such as Materials, Labour, or Miscellaneous activities. To do this you should carry out the following:

Go into **General>Settings>Analysis Groups** and create a group called 'Labour'. You will see three fields (Quote Form, Order Form, Invoice Form). Under each of these is a drop-down list that enables you to define the following options:

- **No Grouping:** All Lines in this Group will be printed
- **Header and Footer:** A Header containing the name of the Analysis Group will be printed followed by all Lines within the Group. At the end of the Group a Footer record will be printed showing a summarised total of the group content
- **Group Totals Only:** A single line will be printed showing the summarised total for the Group

For the purpose of this exercise select the following:

- Quote Form:** Header and Footer
Order Form: No Grouping
Invoice Form: Group Totals Only

Now go into **Inventory>Descriptors** and select 'GENERAL TIME'. In the 'Detail' view select 'Labour' from the drop-down list under field 'Analysis Group'. Do the same against Descriptor 'MISCCHARGE'

Go into **Sales>Sales Orders** and click the 'Add' button. Select Customer 'Jim Gold & Co Ltd'. Click on the 'Create Quote' button (NOT the Create Order button) to generate the Sales Quotation.

Go into the 'Lines' tab and add:

- A line with Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **50**
 A line with Line Type: **Descriptor Code**, Code: **MISCCHARGE**, and Order Qty: **150**

If you go into the 'Detail' view and click on the Quote 'Print' button you will see that each line is

printed along with the 'Header' and 'Footer' of the Analysis Group.

You may wish to go back to [General>Settings>Analysis Groups](#) and amend the setting against Quote Form then reprint the Quote to view the various options

5. Monitoring the Quotation

Various Reports and Views are available to monitor Quotes. Take a look at the following:

Reports: Go to [Sales>Sales Reports](#) and have a look at the following Reports

Sales Quote Listing: Enter 'Expiry Date To' as yesterday to view expired Quotes

Sales Quote Values Chart: Shows Converted, Current and Lost Quote values

Views: Go to [Sales>Sales Views](#) and have a look at the following.

Analysis - Sales Orders: Select 'Order Status' 'Lost' and/or 'Quote'

Customer Analysis with all Orders: Select 'Order Status' 'Lost' and/or 'Quote'

Chart - Open Sales Quotes

Chart - Lost Sales Quotes

Go to [Help>Reference](#) and select [Global Options>Views Analysis Options](#) where you will find guidance as to how the Analysis View can be manipulated to give you the results you want.

6. Converting the Quotation

A Quote can either be converted into a Sales Order or can be flagged as 'Lost'.

6.1. Convert to a Sales Order

Go into the Quote you created above and, in the 'Detail' view click on the 'Convert Quote' button. On the presented panel 'check' the 'Converting to Order' Radio button then click the 'OK' button. The Quote will be immediately converted to a Sales Order with status 'Open'

6.2. Convert to a 'Lost' Status

When converting a Quote into a 'Lost Quote' you are required to select a reason why the Quote was lost. These reasons are user maintained by going into [General>Settings>Quote Lost Reasons](#). Go into that screen and add a couple of reasons to the current list.

Now create a Sales Quote as described in 4.3. In the 'Detail' view click on the 'Convert Quote' button. On the presented panel 'check' the 'Quotation is Lost' Radio button. The field underneath will become active where you can select the reason why it was lost from the drop-down list then click the 'OK' button. The Quote will immediately be given a status of 'Lost'.

6.4 Creating and Monitoring a Sales Order

As stated earlier there are 2 Sales Order Styles used in Ostendo

- Raise Order > Pick Goods > Invoice
- Raise Order > Pick Goods > Despatch > Invoice

We will now go through the Pick Goods to Invoicing process for each option.

1. 'Counter' Sales Order

Go into [Sales>Sale Orders](#) and click the 'Add' button. Select Customer 'Jim Gold & Co Ltd' and Sales Type 'CounterSales' then click the 'Create Order' button. Go into the 'Lines' tab and add

two lines:

Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **5**

Line Type: **Item Code**, Code: **100-2004**, and Order Qty: **1000**

As this is a '**Counter**' Style Sales Order the goods will be picked and an Invoice raised from the picked goods.

1.1. Picking

There are 3 methods by which the Lines can be picked.

Picking from within the line

Go to the '**Lines**' tab in the Order and click on the '**Picked**' tab. Select the specific line in the upper part of the screen and then click the '**Add**' button to create a new pick line in the lower part. Enter the quantity and date of the pick.

- For Stock Items this requires entry of a quantity and a Warehouse/Location. This is defaulted to that held against the Item but can be amended. However, if the Item has 'variants' (Serial Number, Lot Number, Colour, Grade, etc) or you are picking from a different location then place the cursor in the '**Qty**' field and click on the 'Spyglass' Icon that appears in that field. A screen will be presented that shows all the locations of all the variants. You should select the specific variant then click the 'Save' button.
- For non-stock Items (Descriptors, Kitsets, Catalogue Items) it is simply a pick date and quantity

Picking from the Batch Entry bar

Go to the '**Lines**' tab in the Order and click on the '**Picking**' button on the Batch Entry bar. A screen will appear into which you can:

- Click the '**Prefill Pick Lines**' button to add the lines to be picked along with their pick quantities (You can go into the Qty field and amend the quantity if required).
- Click the '**Add**' button to create a new pick line and enter the line that is being picked

Click on the 'Update Sales Order' button to pick the entered values

Auto-Picking from the Batch Entry bar

Go to the '**Lines**' tab in the Order and you should see a button with caption '**Auto Pick**' to the right of the '**Picking**' button on the Batch Entry bar. If you cannot see this then go to **Sales>Settings>Sales Rules** and 'check' both the '**Allow Auto Pick from Counter Orders**' and '**Allow Invoice Creation from Auto Pick**' checkboxes.

Click on the button and another panel will appear asking you to confirm the auto-pick. If you confirm this then all lines will be picked from their default locations. However, if the Item being picked has tacking characteristics such as Serial Number, Grade, etc then the line will NOT be picked and you should carry this out using one of the previous two methods.

If all lines have been picked using this button then, because you also 'checked' the '**Allow Invoice Creation from Auto Pick**' checkbox another small panel will appear asking if you wish to immediately create an Invoice for the picked quantities. If you confirm this then you will be taken through the Invoicing process for this Order..

2. 'Delivery' Sales Order

Go into **Sales>Sale Orders** and click the 'Add' button. Select Customer '**Jim Gold & Co Ltd**' and Sales Type '**DeliveryOrder**' then click the '**Create Order**' button. Go into the '**Lines**' tab and add two lines:

Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **5**
 Line Type: **Item Code**, Code: **100-2004**, and Order Qty: **1000**

As this is a '**Delivery**' Style Sales Order the goods will be picked during the Despatch process and an Invoice raised from the delivered goods.

2.1. Create Sales Delivery

Having created the Sales Order you will now go through the following steps

- Create a Delivery Reference
- Print a Pick List and Pick the Goods
- Add Delivery information
- Despatch the Goods

Go to the '**Detail**' tab in the Order and click on the '**Sales Deliveries**' button. (Alternatively you could go into **Sales>Sales Deliveries** and click the '**Add**' button. On the presented panel leave the 'Radio Button' on '**Create Delivery for a Single Order**' and select the Delivery Order that you created above.)

If you select the '**Lines**' tab you will see all the lines that are required to be 'Picked'

2.2. Print a Pick List and Pick the Goods

Go into the '**Detail**' tab and click on the '**Pick List Status**' Print Icon to produce the Pick List. This list is used to pick the Items, etc and note the Warehouse/Location from where they were picked from along with any sub-level variants (Batch Number, Serial Number, Colour, Size, etc).

Now go back to the Delivery '**Lines**' tab and enter the 'Picked' Lines. You may wish to use the '**Prefill Picked Quantities**' button to automatically enter the 'Required Quantity' in the 'Picked Quantity' field. If you choose this method then you should note that:

- The Item's default Warehouse/Location will be used
- If the Item has sub-level variants then the Picked Quantity field will be zero. You are required to highlight the Line and then click the '**Add**' button and specifically state the variant values and quantities that were picked

2.3. Add Delivery Information

Go back to the '**Detail**' tab and address the following areas:

Freight Information: From the drop-down list select the Shipping Method. (You can create your own shipping methods by going into **Sales>Settings>Customer Shipping Methods**). You can define whether you are going to charge for the freight and whether that charge is taxed.

Weights and Volumes: A Weight and Volume is held against each Item record. Those dimensions are used to evaluate:

- The Planned total Weight and Volume expected from this order.
- The 'Picked' Weight and Volume' for the Items that have been picked

Area is provided for you to enter the actual Weight and Volume for this delivery. Note: The unit of Weight and Volume is user-defined via **File>System Configuration>System Settings**.

Delivery Details: If you click on the '**Additional Delivery Address Details**' button you can:

- Use the default address that is held against the Customer, or
- Select an alternative Delivery Address from the drop-down list, or
- Simply key in a Delivery Address

Picking Information: You can print a Pick List by clicking on the printer icon adjacent to the '**Pick**

List Status field. Once the Items have been picked you can update this screen by clicking the 'Picked By' drop-down list and selecting the person who picked to goods.

Shipping Labels: Enter the number of Pieces then print the Shipping Labels

Delivery Confirmation: Print the Delivery Document and confirm the Date that it was delivered.

Finally, click the '**Update Delivery Status to Shipped**' button. This will activate the '**Sales Invoices**' Button. This, along with other Invoicing methods, will be covered in the next section.

3. Invoicing

We will now go through the various options available within Sales Invoicing. These can be generated from five sources:

- **Invoices from 'Counter' Sales Order** - Once an Order has been 'picked' the picked lines are ready for Invoicing via the 'Counter Order' Detail screen or a 'Batch Invoicing' routine
- **Invoices from 'Delivery' Sales Order** - Once an Order has been 'delivered' the delivered lines are ready for Invoicing via the 'Sales Delivery' screen or a 'Batch Invoicing' routine
- **Invoice Schedule** - A schedule of Invoices can be defined. This can cover - for example - a Service Schedule and consists of a fixed schedule with pre-defined Invoices. The Invoice generation is carried out via a '**Batch Invoicing**' routine
- **Recurring Invoices** - A repeating Invoice based on a regular - and open ended - frequency. The Invoice generation is carried out via a '**Batch Invoicing**' routine
- **Direct Invoices** - You can create an Invoice 'On the fly' within its own routine

For the first 4 Invoice Sources you have the option to print them on a 1 for 1 basis (One Invoice for each 'delivery') or in batch where you can combine them into a single 'consolidated' Invoice.

3.1. Create an Invoice for a 'Counter' Type Sales Order

Recall the Sales Order that you created in Section 5.1. and go to the Order '**Detail**' screen. Click on the '**Sales Invoices**' button. This will bring up a panel prefilled with this Sales Order information. Click on the '**Create**' button. An Invoice will be created for the deliveries you made above. You should note the following:

- You can amend the Invoice Date prior to printing. The amended date cannot be earlier than the '**Customer Invoicing Date**' defined in **File>Financial Configuration>Cutoff Dates**.
- You can 'check' the '**Don't Print on Statement**' checkbox. This is useful for if you have made a mistake during Invoicing and have created an (Internal) Credit. Both the originating Invoice and the Credit can be flagged not to appear on the Customer Statement.
- You can amend any Freight Charges that are included with the Invoice

Click on the '**Print**' icon to print the Invoice. For now just '**Preview**' the Invoice.

Go into the '**List**' view and delete this Invoice (We will be creating it later along with other Invoices later in these exercises)

3.2. Create an Invoice for a 'Delivery' Type Sales Order

Go to **Sales>Sales Deliveries** and locate the Sales Delivery that you created in Section 2.1. and go to the Delivery '**Detail**' screen. Click on the '**Sales Invoices**' button. This will bring up a panel prefilled with this Sales delivery information. Click on the '**Create**' button. An Invoice will be

created for the **'Deliveries'** you made via the Sales Deliveries screen. Click on the **'Print'** icon to print the Invoice. For now just **'Preview'** the Invoice.

Go into the **'List'** view and delete this Invoice (We will be creating it later along with other Invoices later in these exercises)

3.3. Create an Invoicing Schedule without a Sales Order

This covers a planned schedule of pre-defined Invoices. This is useful where you have an agreed schedule for work carried out (Example: You carry out daily/weekly garden maintenance but only Invoice on a monthly basis)

Go into **Sales>Recurring Invoices** and click on the **'Add'** button. On the presented screen you can create a schedule of planned Invoices as follows:

- Select a Customer from the drop-down list against field **'Customer'** then
- Select a **'Contract Style'** of **'Fixed Term'**
- In the **'Invoice Line Details'** section down the left of this screen define:
 - Frequency of the Invoicing Schedule
 - Number of Times that it will occur
 - The Start Date
 - The nominal amount per Invoice
- In the **'Invoice Line Details'** section down the right of this screen define:
 - The **'Descriptor Code'** that will be printed on the Invoice
 - Amend the Description to suit your requirements
 - Apply a Tax Code if required
- Click the **'Create'** Button to generate the schedule

You can view the schedule by going to the **'Scheduled Invoices'** tab

If you go to the **'Detail'** tab you can enter additional information that will appear on each generated Invoice. The generation of the actual Invoice will be described in Exercise 5.4.5.

3.4. Create a Recurring Invoice

This covers generation of Invoices based on a defined frequency where the next **'Planned'** Invoice is only scheduled relative to the Actual date the previous Invoice was printed.

Go into **Sales>Recurring Invoices** and click on the **'Add'** button. On the presented screen you can create a schedule of planned Invoices as follows:

- Select a Customer from the drop-down list against field **'Customer'** then
- Select **'Contract Style'** of **'Continuous'**
- In the **'Invoice Line Details'** section down the left of this screen define:
 - Frequency of the Recurring Invoicing
 - The Start Date
 - The nominal amount per Invoice
- In the **'Invoice Line Details'** section down the right of this screen define:
 - The **'Descriptor Code'** that will be printed on the Invoice
 - Amend the Description to suit your requirements
 - Apply a Tax Code if required
- Click the **'Create'** Button to generate the first Planned Invoice.

You can view the generated Invoice by going to the **'Scheduled Invoices'** tab

If you go to the **'Detail'** tab you can enter additional information that will appear on each generated Invoice. The generation of the actual Invoice will be described in Exercise 5.4.5.

3.5. Batch Invoice Creation

This section covers an alternate way of creating Invoices. It also provides facility to combine 'Invoices' from all sources into a single Invoice by Customer.

To prepare for creating a combined invoice you should first go into **Sales>Customers** and select the Customer you created in Exercise 1. then click on the '**Pricing and Invoicing**' tab. In the displayed panel you will see a field '**Invoice Consolidation**'. It has two options:

- **Consolidate by Order**: All Invoicable transactions for the same Order are combined into a single Invoice.
- **Consolidate by Customer**: All Invoicable transactions for the same Customer are combined into a single Invoice.

For the purpose of this exercise select '**Consolidate by Customer**'

Now, go into **Sales>Batch Invoicing** and select the '**Add**' button. A screen is presented in which you should 'check' the following

- Include Sales Orders (Counter Sales)
- Include Sales Order Deliveries
- Include Recurring (Contract) Invoicing
- Select you Customer in the drop-down in field '**Specific Billing Customer**'
- Click the '**Create**' button.

The program will now gather all the Planned Invoices from the Sales activities you entered above and combine them into a single Invoice for your Customer.

For the generated Invoice go to the '**Detail**' screen and click on the '**Invoice Print**' button and select '**Preview**' option. We will print the Invoice '**In Batch**' later in this session.

3.6. Direct Invoice

There are instances where you may wish to create an Invoice (or Credit) where there is no prior Sales Order. To do this, go into **Sales>Direct Invoicing** and click on the '**Add**' button. On the displayed panel 'check' the '**Create an Invoice or Credit without an Order**' radio button and select the Customer you created above from the drop-down against field '**Billing Customer**' then click the '**Create**' button. Enter the Invoice details as required and then click the print '**Print**' button. This time 'uncheck' the '**Preview**' checkbox and print the Invoice back to your screen.

3.7. Create a Credit by reversing an existing Invoice

To create a Credit go into **Sales>Direct Invoicing** and 'check' the '**Create Credit by reversing an existing Invoice**' radio button. In the drop-down against field '**Existing Invoice Number**' select the Invoice that you are Crediting then click the '**Create**' button. Enter the Credit details as required and then click the print '**Print**' button. 'Uncheck' the '**Preview**' checkbox and print the Credit back to your screen. This action will create an exact reversal of the selected Invoice

3.8. Batch Invoice Printing

Up to now you have seen that you can print individual Invoices as they are created. But you can also print previously created - but not printed - Invoices in Batch. Go to **Sales>Batch Invoice Printing** and select your Customer from the drop-down against field '**Specific Customer**' then press the '**OK**' button. On the presented screen you can:

- Select a specific Invoice and click the '**Print Current Invoice**' button, or
- Select multiple Invoices and then click the '**Print All Selected Invoices**' button

You should carry out the second option. I.e. Select all the displayed Invoices, 'check' the '**Preview**' checkbox then click the '**Print All Selected Invoices**' button.

On the generated Invoice you will see that it creates a combination of all unprinted Invoices from the selection criteria. Each included Invoice is itemised along with its details

4. Customer Payments

Customer payment covers:

- Deposits
- Invoice Payments
- Raising Credit Notes
- Matching Payments, Deposits, Credits to Invoices
- Banking the payments

4.1. Customer Deposits

You may receive a deposit from a Customer prior to raising an Invoice (Example: When the time the order was placed). This deposit should subsequently be matched against an Invoice that has been raised against the Customer.

To create a Deposit go to **Sales>Customer Deposits** then click the **'Add'** button. On the presented screen carry out the following:

Customer: From the drop-down list select the Customer that you created above

Deposit for: Select **'UnAllocated'** from the drop-down list

Payment Method: Select **'Cheque'**

Payment Reference: Enter a reference (Example: **Deposit**)

Banking Method: Select **'Using Bank Deposit'** from drop-down list

Account for Payment: Select **'MyBank'** from the drop-down list. If you don't have an account in the drop-down list then go to **File>Financial Configuration>Payment**

Accounts and create an Account by entering (say):

Code: **Trading**

Description: **Trading Account**

Cost Centre: Select **'BANK'** from the drop-down list

Deposit Values: Enter a deposit amount

Note: If the Customer's payment details (Payer Name, Bank Name, Branch Name, Payment Method) are generally the same then you can add these details to the Customer record (**Sales>Customers** and click on the **'Payment'** tab). Whenever you create a Customer Deposit or Payment the records are prefilled with that information. You can, of course, amend them against the specific payment

Now **'Save'** the record. You may wish to print the **'Customer Deposit'** report (**Sales>Reports>Customer Deposit Listing**).

The next step is to match the Deposit against an Invoice.

Go into **Sales>Customer Payments** then click the **'Add'** button. On the presented screen select the above Customer then go to field **'Payment Style'** and select **'Use Customer Deposit'** from the drop-down list. A field will be presented to the right of this field from where you should now select the above 'Deposit'.

Now click on the **'Apply to Invoices'** tab at the top of the screen and match the Deposit to outstanding Invoices by either clicking the **'Auto Apply'** button or going into the **'Applied Amount'** field against selected Invoices and applying a specific amount.

4.2. Customer Payments

Upon receipt of a Payment from a Customer you should enter that receipt into Ostendo and then

match it against outstanding Invoice(s). To create a Payment record, go to **Sales>Customer Payments** then click the **'Add'** button. On the presented screen carry out the following:

Customer: From the drop-down list select the Customer that you created above

Payment Style: Select **'Received Payment'**

Payment Method: Select **'Cheque'**

Payment Reference: Enter a reference (Example: **Invoice Payment**)

Banking Method: Select **'Using Bank Deposit'** from drop-down list

Account for Payment: Select **'MyBank'** from the drop-down list. If you don't have an account in the drop-down list then go to **File>Financial Configuration>Payment Accounts** and create an Account by entering (say):

Code: **Trading**

Description: **Trading Account**

Cost Centre: Select **'BANK'** from the drop-down list

Payment Values: Enter the payment amount

Now **'Save'** the record. You may wish to print the Customer Payment Report by clicking on the **'Reports'** button to the right of the screen.

The next step is to match the Payment against an Invoice. On this same screen click on the **'Apply to Invoices'** tab at the top of the screen and match the Payment to outstanding Invoices by either clicking the **'Auto Apply'** button or going into the **'Applied Amount'** field against selected Invoices and applying a specific amount.

4.3. Raising Credit Notes

To create a Credit Note you should go into **Sales>Direct Invoicing** and click on the **'Add'** button. You can see on this screen that Credits can be processed in three ways:

- Create a Credit Note and subsequently match to an Invoice
- Reverse an existing Invoice and then subsequently match as 'Payment'
- Reverse an existing Invoice and immediately match as 'Payment'

4.3.1. Create a Credit Note for subsequent matching to an Invoice

To create a Credit Note without reference to a previous Invoice 'check' the radio button **'Create an Invoice or credit without an Order'** and (under **'Billing Customer'**) select the Customer you used in section 5.4. Click the **'Create'** button and complete the **'Detail'** screen. On the **'Lines'** tab enter the Credit using a Descriptor Code where the value is the Credit amount and the Quantity is **-1** (Minus 1). If you now go back to the **'Detail'** tab and click on the **'Print'** button and bring back to your screen you will see that the negative amount will change the 'Invoice' title to **'Credit'**. You can now match off the Credit as described in 5.5.4.

4.3.2. Reverse an existing Invoice and then subsequently match as 'Payment'

To reverse an existing Invoice you have two options:

- Highlight the Invoice on the Batch Invoicing **'List'** screen (**Sales>Batch Invoicing**) and click the **'Credit'** button to the right of the screen
- Go to the Direct Invoicing screen (**Sales>Direct Invoicing**) then click the **'Add'** button. On the displayed screen 'check' the radio button **'Create credit by reversing an existing Invoice'** and (under **'Existing Invoice Number'**) select an Invoice to be reversed. Click the **'Create'** button.

In both the above instances Ostendo will create a Credit by duplicating the selected Invoice with a reversed quantity. If, in the created Credit's Main screen, you click on the **'Print'** button you will see that the negative amount will change the 'Invoice' title to **'Credit'**. You can now match off the Credit as described in 5.5.4.

4.3.3. Reverse an existing Invoice and immediately match as 'Payment'

To reverse an existing Invoice 'check' the radio button '**Create credit by reversing an existing Invoice**' and (under '**Existing Invoice Number**') select an Invoice to be reversed. You should also 'check' the '**Apply the Credit as a Payment**' checkbox. Click the '**Create**' button and Ostendo will create a Credit by duplicating the selected Invoice with a reversed quantity. If you now print the Credit clicking the '**Print**' button then the status of both the originating Invoice and reversing Credit will be flagged as '**Fully Paid**'

4.4. Matching Payments, Deposits, Credits to Invoices

To match a Credit Note to an Invoice go to **Sales>Customer Payments** then click the '**Add**' button. On the presented screen carry out the following:

Customer: From the drop-down select the Customer against whom the above Credit was raised.

Payment Style: Select '**Match Credits to Invoices**'

Credit Number: Select the Credit that you created above

The next step is to match the Credit to an Invoice. On this same screen click on the '**Apply to Invoices**' tab at the top of the screen and match the Credit amount to outstanding Invoices by either clicking the '**Auto Apply**' button or going into the '**Applied Amount**' field against selected Invoices and applying a specific amount. **Note:** The Credit Note will also be displayed in the main panel BUT, because it is the main subject of the matching process, it should not take part in the matching process. Therefore, for this line do NOT 'check' the 'Select' checkbox against this line

Having allocated the Credit to Invoice(s) click on the '**Update Selected Invoices**' button at the bottom of the '**Lines**' screen. The status of both the Credit and each selected Invoice will be updated to state whether they have been **partially applied** or **fully applied**.

4.5. Bank Deposits

There are two options to placing Payments and/or Deposits into a Bank Account.

- Pay each payment receipt directly into the Bank Account
- 'Batch' the payment receipts and use a single Deposit Slip to pay into the Bank Account.

4.5.1. Pay directly into a Bank Account

If, in the Deposit or Payment Receipt you select '**Directly to Account**' in '**Banking Method**' then any Payment or Deposit will be posted directly to the Cost Centre linked to the Payment Account

4.5.2. Pay into Bank Account using a Deposit Slip

If you select '**Using Bank Deposit**' in '**Banking Method**' then a two-step process is used.

- The Payment/Deposit will initially be posted to an '**Un-Deposited Funds**' Cost Centre
- The '**Un-Deposited Funds**' will be 'batched' and posted to the '**Payment Account**'

Step 1: When creating the Deposit or Payment record select '**Using Bank Deposit**' in '**Banking Method**' and process the payment or deposit as outlined above

Step 2: Go into **Sales>Bank Deposits** and click the '**Add**' button to create a new Bank Deposit.

- Select the Bank ('**Payment Account**') from the drop-down list
- Click on the '**Payments Selection**' tab to show all payments/deposits scheduled to be deposited into the selected Bank Account
- Select the Payments/Deposits to be included in this Bank Deposit batch
- Click the '**Update Selected Payments**' button at the bottom of the screen.
- Go to the '**Detail**' tab and print the Deposit Slip

Having deposited the funds you should go back to **Sales>Bank Deposits** for this Deposit and click the **'Update Status to Banked'** button.

5. Aging and Statements

You should first define the aging periods to be used with both the **Customer Aging report** and Customer Statements. Go to **File>Financial Configuration>Aging Periods** and **File>Financial Configuration>Statement Periods**. In both instances you can define if the aging periods are based on the **Invoice Date** or the payment **Due Date**.

You can now produce the documents via:

Customer Aging: **Sales>Customer Aging**, or
Customer Statement: **Sales>Customer Statements**

6.5 Creating a One-Step Order - no Backorder

This style of Sales Orders allows you to raise the order, pick the goods, invoice what you have picked and (optionally) take payment - all in one screen. This has been designed for the environment where the Order is taken and the Invoice printed. The Invoice effectively becomes a Picklist and the goods are picked. If the Goods are found not to be in Inventory then the Invoice can be regenerated to reflect what has been picked. Upon completion the Sales Order is automatically 'closed'.

1. Preparation

1.1. Sales Type

The first step is to create a Sales Type that tells Ostendo that it is a One-Step Sales order. Go into **Sales>Settings>Sales Types** and click the 'Add' button. Add a Sales Type record with the following content:

Sales Type: **OneStep1**
Description: **One Step with No Backorder**
Creation Status: **Open**
Numbering Is: **Automatic**
Prefix: **OS**
Sales Style: **Counter**
Rental Order: **Leave Blank**
Invoicing Style: **One Step No Backorder**

1.2. Invoice Number Style

The next (optional) step is to tell Ostendo that the Invoice Numbering should equate to the Sales Order Number. Therefore go into **Sales>Settings>Sales Rules** and 'check' the checkbox against **'Use Order No as Invoice No for One Step'**

2. Creating the Order

Go into **Sales>Sale Orders** and click the 'Add' button. Select Customer **'Jim Gold & Co Ltd'** and Sales Type **'OneStep1'** then click the **'Create Order'** button. Go into the **'Lines'** tab and add two lines:

Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **5**
Line Type: **Item Code**, Code: **100-2004**, and Order Qty: **1000**

2.1. Pick and Invoice

Now you can pick and Invoice the Order. The following methods can be used. You may wish to do each method in turn – creating a new Order as above in each case

2.1.1. Key in **Ctrl-I**. This action will:

- Pick the Lines
- Create the Invoice
- Print the Invoice and update its status to 'Printed'
- Update the Sales Order status to 'Completed'

2.1.2. Click on the Invoice button and 'check' the 'Preview' checkbox then output the Invoice to the screen, printer, or email as required. This action will:

- Pick the Lines
- Create the Invoice with status 'Planned' for later printing via 'Batch Invoice Printing'
- Update the Sales Order status to 'Completed'

2.1.3. Click on the Invoice button and 'uncheck' the 'Preview' checkbox then output the Invoice to the screen, printer, or email as required. This action will:

- Pick the Lines
- Create the Invoice
- Print the Invoice and update its status to 'Printed'
- Update the Sales Order status to 'Completed'

3. Amending the Order

Of course the situation will arise where, for various reasons, the Invoiced Quantities could not be picked from stock. This requires you to amend the previously picked and Invoiced quantities. Therefore go into the Sales Order screen (**Sales>Sales Orders**) and 'check' the '**Include Closed Status**' checkbox at the bottom of the screen.

Locate the above '**Closed**' Order and click on the '**Lines**' tab. Simply amend the Order quantity and print the Invoice using one of the three options described above. Ostendo will create adjustment records for the difference between the old and the new quantities; plus create a revised Invoice as described below

3.1. Original Invoice was not printed

If the original Invoice was not printed (as 2.1.2. above) then the Invoice will simply be replaced with a revised Invoice using the same Invoice Number

3.2. Original Invoice was printed

If the original Invoice was printed (as 2.1.1. and 2.1.3. above) then the Invoice will be reversed in its entirety and a new Invoice generated.

3.2.1. If the Invoice Numbering Style (as set in 1.2. '**Use Order No as Invoice No for One Step**' above) is NOT checked then standard Invoice numbering will be used. Assuming that the Original Invoice was INV12345 for \$100 then Ostendo will take the next available Invoice Number and fully credit it

Here is an example of what Ostendo will create

Original Invoice	IN12345 for \$100
Credit Invoice	IN12454 for -\$100
New Invoice	IN12455 for \$120

3.2.2. If the Invoice Numbering Style (as set in 1.2. '**Use Order No as Invoice No for One Step**'

above) is checked then the Invoice numbering will comprise of the Order Number followed by a suffix. Assuming that the Order Number was SO33445 then the generated Invoices will be

Original Invoice	SO33445-1 for \$100
Credit Invoice	SO33445-2 for -\$100
New Invoice	SO33445-3 for \$120

4. Payments and/or Deposits

In addition to the One-Step process to generate the Invoice you can also include a Deposit or Payment if required. To demonstrate this, create another Sales Order and add a couple of lines as described in 2 above. Go back to the 'Detail' screen of the Sales order and click on the 'Order Deposit' tab. If you click on the 'Create Deposit' button you can enter details of the Deposit. For the purpose of our exercise create a Deposit of (say) \$200 and click the 'Accept' button.

If you go to *Sales>Customer Deposits* you will see that the Deposit record has been generated.

Go back to the 'Order Deposits' in the Sales Order and 'check' the 'Auto Apply Deposit' checkbox. If you now issue the Lines and produce the Invoice as described in 2.1 you will see that the Deposit amount is shown in the printout. You will also find that the Deposit has been applied to the Invoice. To see this go into *Sales>Customer Payments* where you will see the Deposit has been applied to the Invoice. If you cannot see the Payment record then it has been fully applied and you can 'check' the 'Include Fully Applied Status' checkbox at the bottom of the List screen.

6.6 Creating a One-Step Order with Backorders

Order allows you to raise the order, pick the goods, invoice what you have picked and (optionally) take payment - all in one screen. This has been designed for the environment where the Order is taken and the Invoice printed. The Invoice effectively becomes a Picklist and the goods are picked. If the Goods are found not to be in Inventory then the Invoice can be regenerated to reflect what has been picked. Upon completion, if the Sales Order Lines are fully picked then the Sales Order is automatically 'closed'. If the Order Lines are not fully picked then the Sales Order status will be updated to 'In Progress' and the Lines adjusted to reflect the undelivered quantities.

1. Preparation

1.1. Sales Type

The first step is to create a Sales Type that tells Ostendo that it is a One-Step Sales order. Go into *Sales>Settings>Sales Types* and click the 'Add' button. Add a Sales Type record with the following content:

Sales Type: **OneStep2**
Description: **One Step with Backorder**
Creation Status: **Open**
Numbering Is: **Automatic**
Prefix: **OS**
Sales Style: **Counter**
Rental Order: **Leave Blank**
Invoicing Style: **One Step With Backorder**

1.2. Invoice Number Style

The next (optional) step is to tell Ostendo that the Invoice Numbering should equate to the Sales Order Number. Therefore go into *Sales>Settings>Sales Rules* and 'check' the checkbox against 'Use Order No as Invoice No for One Step'

2. Creating the Order

Go into **Sales>Sale Orders** and click the 'Add' button. Select Customer '**Jim Gold & Co Ltd**' and Sales Type '**OneStep2**' then click the '**Create Order**' button. Go into the '**Lines**' tab and add two lines:

Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **5**

Line Type: **Item Code**, Code: **100-2004**, and Order Qty: **1000**

2.1. Pick and Invoice

Now you can pick and Invoice the Order. The following methods can be used. You may wish to do each method in turn – creating a new Order as above in each case

2.1.1. Key in **Ctrl-I**. This action will:

- Pick the Lines
- Create the Invoice
- Print the Invoice and update its status to 'Printed'
- Update the Sales Order status to 'Completed'

2.1.2. Click on the Invoice button and 'check' the 'Preview' checkbox then output the Invoice to the screen, printer, or email as required. This action will:

- Pick the Lines
- Create the Invoice with status 'Planned' for later printing via 'Batch Invoice Printing'
- Update the Sales Order status to 'Completed'

2.1.3. Click on the Invoice button and 'uncheck' the 'Preview' checkbox then output the Invoice to the screen, printer, or email as required. This action will:

- Pick the Lines
- Create the Invoice
- Print the Invoice and update its status to 'Printed'
- Update the Sales Order status to 'Completed'

3. Amending the Order

Of course the situation will arise where, for various reasons, the Invoiced Quantities could not be picked from stock. This requires you to amend the previously picked and Invoiced quantities. Therefore go into the Sales Order screen (**Sales>Sales Orders**) and locate the above Order and click on the '**Lines**' tab. Note: If the Order has been fully completed then the status will be '**Closed**'. In this case 'check' the '**Include Closed Status**' checkbox at the bottom of the screen.

In the '**Lines**' screen of the Sales Order you will see a '**One Step Invoicing Mode**' bar. Select '**Adjust Previous Invoice**' from the drop-down list to get Ostendo to display the quantity Issued against the previous Invoice. For each line where an adjustment is required select the line and, in the lower part of the screen, adjust the Quantity by entering the quantity to be invoiced in field '**Invoice Qty**'. If you now print the Invoice using one of the three options described above Ostendo will create adjustment records for the difference between the old and the new quantities; plus create a revised Invoice as described below

3.1. Original Invoice was not printed

If the original Invoice was not printed (as 2.1.2. above) then the Invoice will simply be replaced with a revised Invoice using the same Invoice Number

3.2. Original Invoice was printed

If the original Invoice was printed (as 2.1.1. and 2.1.3. above) then the Invoice will be reversed in its entirety and a new Invoice generated.

3.2.1. If the Invoice Numbering Style (as set in 1.2. **'Use Order No as Invoice No for One Step'** above) is NOT checked then standard Invoice numbering will be used. Assuming that the Original Invoice was INV12345 for \$100 then Ostendo will take the next available Invoice Number and fully credit it

Here is an example of what Ostendo will create

Original Invoice	IN12345 for \$100
Credit Invoice	IN12454 for -\$100
New Invoice	IN12455 for \$120

3.2.2. If the Invoice Numbering Style (as set in 1.2. **'Use Order No as Invoice No for One Step'** above) is checked then the Invoice numbering will comprise of the Order Number followed by a suffix. Assuming that the Order Number was SO33445 then the generated Invoices will be

Original Invoice	SO33445-1 for \$100
Credit Invoice	SO33445-2 for -\$100
New Invoice	SO33445-3 for \$120

4. Backorders

If all lines are fully picked then the Sales Order status will be updated to 'Completed' and no backorder created.

If ANY line is not fully picked then the Sales Order status will remain as **'In Progress'** and the Lines in the Sales Order will show the outstanding Order Quantity. As before, you can create an Invoice for this backorder in the same manner described above.

5. Payments and/or Deposits

In addition to the One-Step process to generate the Invoice you can also include a Deposit or Payment if required. To demonstrate this, create another Sales Order and add a couple of lines as described in 2 above. Go back to the **'Detail'** screen of the Sales order and click on the **'Order Deposit'** tab. If you click on the **'Create Deposit'** button you can enter details of the Deposit. For the purpose of our exercise create a Deposit of (say) **\$200** and click the **'Accept'** button.

If you go to **Sales>Customer Deposits** you will see that the Deposit record has been generated.

Go back to the **'Order Deposits'** in the Sales Order and 'check' the **'Auto Apply Deposit'** checkbox. If you now issue the Lines and produce the Invoice as described in 2.1 you will see that the Deposit amount is shown in the printout. You will also find that the Deposit has been applied to the Invoice. To see this go into **Sales>Customer Payments** where you will see the Deposit has been applied to the Invoice. If you cannot see the Payment record then it has been fully applied and you can 'check' the **'Include Fully Applied Status'** checkbox at the bottom of the List screen.

6.7 Creating and Monitoring a Rental Order

Rentals include two basic areas

- The Sales Order and Billing function
- The Equipment Availability and Tracking function

In Ostendo the Billing function is covered using Descriptors. This Descriptor is linked to an Item (Equipment). When generating a Rental Order (which contains both a Start and anticipated Return

Date) the Descriptor is added to the Order Line. This then allows Ostendo to display the linked Item's planned or actual availability. Billing will be carried out using Ostendo's current pricing structure held against the Descriptor

1. Setup

1.1. Rental Items

You should first identify the Rental Item by going into *Inventory>Items* and creating a standard Item Code (For example: **Wheelbarrow**)

1.2. Descriptors

The next step is to identify the Descriptors and their Charge Rate. For example the above Wheelbarrow may have 3 charges based upon the Time Period of the Rental.

Descriptor Code	Unit	Unit Price
Wheelbarrow – Hour	Hour	\$12.00
Wheelbarrow – Day	Day	\$30.00
Wheelbarrow – Week	Week	\$70.00

Go into *Inventory>Descriptors* and create each of the above Descriptors. At the same time you will see a 'tab' on the Descriptor's Main screen called '**Rental Settings**'. Go into this tab and enter:

Rental Descriptor: 'Check' the checkbox to denote that it is a Rental Descriptor.

Rental Item Code: From the dropdown list select the above '**Wheelbarrow**' Item

Default Rental Quantity: This represents the Item quantity normally rented when the Descriptor is used. Leave this as **1**

1.3. Sales Types

The last setup step is to define a Sales Type. Go into *Sales>Settings>Sales Types*. And create a Sales Type '**Rental Order**'. Identify it as being a Rental Order by 'checking' the '**Rental Order**' checkbox.

2. Creating the Rental Order

2.1. Order Creation

Go into *Sales>Sales Orders*. and click the '**Add**' button to create a new order (use the above '**Rental Order**' Sales Type). You will see on the Order creation panel a **Rental Start** and **Rental End** date (prefilled with system date). Amend these dates as required to conform to the Rental period. Now click on the **Create Order** button to generate the Rental Order.

Go to the Sales Order Line and add the above 'Rental' Descriptor. In the order line you will see two additional features specifically for Rental Lines

2.1.1. An additional 'Tab' is available in the lower part of the screen called '**Rental Code**'. This contains the linked Item Code and its Description.

2.1.2. The '**Info Bar**' across the centre of the screen shows the current and available stock against the linked Item. If you click on this button the Item's availability screen will appear. If you click on the '**Projected Availability**' tab you will see that the Item's availability is taking into account Rental Issues, and Rental returns.

Note: The Issue and Return dates are those entered and maintained against the Sales Order

2.2. Order Monitoring and Maintenance

In the Sales Order **'Detail'** screen you will see a **'Retail Info'** tab. In that panel the following fields are present:

Rental Start Date: You can amend this as required. This will be reflected in the availability status of Rental Line Items

Rental End Date: You can amend this as required. Again, this will be reflected in the availability status of Rental Line Items

Rental Items Issued: 'Check' this checkbox if you have issued the Item to the Customer. This updates this Order header and provides a status relating to the whereabouts of the rented Items

Rental Items Returned: 'Check' this checkbox if the Item(s) have been returned from the Customer. This will then remove this line from the Rental lines in the Inventory Availability screens

7 6. Assembly Orders

The Assembly Order process covers manufacturing of a Product from components. Upon completion the Product is moved into Stock. Assembly Orders can be defined to run with a level of complexity required by the User. For example:

	<u>Bill of Material</u>	<u>Routing</u>	<u>Resources</u>
Assemble 'On the fly'	-	-	-
Use Bills of Material	Y	-	-
Use Routing Only	-	Y	-
BOM & Routing	Y	Y	-
Routing with Resources	-	Y	Y
BOM & Routing & Resources	Y	Y	Y

7.1 Preparation

In preparation for commencing with Assembly Orders you should refer to the following

Assembly Rules: This allows the Systems Administrator to nominate rules that will apply to Assembly Orders. Go into [Assembly>Settings>Assembly Rules](#) and set the rules to your requirements

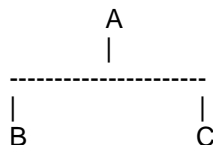
Step Names: Within a BOM you can include a series of 'Steps' that define what activities to carry out when making the Finished Item. If you go into [Assembly>Settings>Step Names](#) you will notice that one step is annotated as the 'Default' step. Whenever a new BOM is created it is always given this Default. At that time you can replace this with any other Step, but a BOM must have at least one Step.

7.2 Bill Of Material

A Bill Of Material is a pre-defined list of Items, Descriptors and Labour Codes that are used to make the required Parent Item.

1. Creating a BOM

To demonstrate Bills Of Material we will begin by creating a BOM with the following Structure



Go into [Inventory>Items](#) and 'Add' 3 Items

<u>Item Code</u>	<u>Description</u>	<u>UOM</u>
A	Parent	Each
B	Component B	Each
C	Component C	Each

Now go into [Assembly>Bills of Material](#) and click the 'Add' button. On the presented Panel enter A into field 'Item Code' then click the 'Create' button to display the Bills of Material 'Detail' panel

Two fields of note (for now):

BOM Batch Size: This represents the quantity that will be made when the 'Lines' of the BOM are used. I.e. The quantities entered into the BOM Lines will make this quantity of the Parent Item. Enter a Quantity of **10** in this field

Quantity for setup Cost Ratio: A Setup activity is normally carried out once per Assembly Order. For costing purposes the cost of this Setup should be apportioned across the quantity being made to arrive at the 'Setup' Cost per finished Item. This field is available to annotate a 'nominal' quantity used for this purpose. It is prefilled with 1 but you should amend this to reflect the 'Average' Order Quantity. We will address this later but, for now, enter a quantity of **20** in this field.

Routing: Firstly let us address the Routing. Across the centre of the screen you will see a series of tabs. Click on tab '**Routing Steps**'. You will see that this BOM has a single Routing Step '**Assembly**'. Click on the '**Add**' button and add a new Step (say) '**QA**'. Go to the '**Instructions**' field and insert some simple Instructions.. You should now have two Steps in the Routing.

Step **10** **Assembly**
Step **20** **QA**

Lines: Now click on the '**Lines**' tab at the top of the screen. This will contain the lines that will be used to make the Assembly. Add the following Lines

<u>Line</u>	<u>Line Type</u>	<u>Code</u>	<u>Qty</u>
10	Item Code	B	10
20	Item Code	C	20

This covers the material requirements to make **10** of the Parent Item (I.e. to cover BOM Batch Size entered above)but what about the Labour hours used to put it together?

There are two methods by which '**Labour**' can be attached to a BOM. I.e. Use a '**Descriptor**' or a '**Labour Code**'. If you intend to record Time Bookings by **Employee** then you would use a '**Labour Code**' (This is covered in more detail later). For now we will add a '**Descriptor**'. Therefore click on the '**Add**' Button and add the following:

<u>Line</u>	<u>Line Type</u>	<u>Code</u>	<u>Qty</u>
30	Descriptor Code	GENERAL TIME	15

This will mean that each Parent Item will require 1½ hours to assemble. I.e. A total of 15 Hours to produce a BOM Batch Size of 10.

Add another Line as follows

<u>Line</u>	<u>Line Type</u>	<u>Code</u>	<u>Qty</u>
40	Descriptor Code	GENERAL TIME	1

In this instance go to the last line in the lower part of the screen and select '**Step**' '**QA**' You have now denoted that the process requires 0.1 hour to carry out a QA check against each Item.

'**Add**' another Line as follows

<u>Line</u>	<u>Line Type</u>	<u>Code</u>	<u>Qty</u>
50	Descriptor Code	GENERAL TIME	4

As above, go to the last line in the lower part of the screen and select '**Step**' '**QA**' Additionally, go to the field '**Run or Setup**' and select '**Setup**'

You have now stated that that the process requires 4 Hours to set up this Step in preparation for carrying out the 'Run' activity defined by Line 40

2. Phantoms

Go to the 'Detail' tab where you will see a checkbox, which if 'checked', will enable this BOM to be become a 'Phantom' when used in a higher assembly. To explain this let us assume that this Assembly is 'Butter'. If you created an Assembly Order for 'Butter' you would use 'Cream' as the Component (or Ingredient).

We could also have a Bill of Material called 'Cake' and it requires 'Butter' as one of its components (or ingredients).

Under normal circumstances if you raised an Assembly Order for 'Cake' then the Ingredient 'Butter' would be required. However, if the 'Phantom' checkbox is 'checked' then an Assembly Order for 'Cake' would state a requirement for 'Cream' taking into account the quantity of 'Butter' required.

2.1. 'Phantoms' In Action

If you wish to see this in action then go into *Inventory>Items* and add the following Item

<u>Item Code</u>	<u>Description</u>	<u>UOM</u>
D	Raw Material	Each

Now go into *Assembly>Bills of Material* and click the 'Add' button. On the presented Panel enter C into field 'Item Code' then click the 'Create' button to display the 'Bills Of Material' 'Detail' panel. In that panel 'check' the 'Phantom Setting' checkbox. Now click on the 'Lines' tab at the top of the screen and add the following Lines

<u>Line</u>	<u>Line Type</u>	<u>Code</u>	<u>Qty</u>
10	Item Code	D	2
20	Descriptor Code	GENERAL TIME	1

Finally, go into *Assembly>Assembly Orders* and create an Order with quantity 1 of Item A. In the generated Order ('Lines' tab) you will see that Item D is displayed with quantity of 4 (i.e. The quantity of 2 of C required in making 1 of A multiplied by the quantity 2 of D required to make 1 of C)

3. Scheduling

Across the centre of the BOM 'Detail' tab you will see fields relating to Leadtime, Setup, and Run Times. These fields are used when creating the Assembly Order as follows:

Leadtime: When creating an Assembly Order (as described in exercise 6) you will see a field called 'Required Date'. The Leadtime days are subtracted from that date to arrive at a 'Start Date'. (i.e. The calculated date or the current date whichever is the later).

Setup Duration: The setup duration is applied to the created order using the calendar defined in *General>Settings>Work Times*.

Run Duration: The run duration is then added to Setup time taking into account the Ordered Quantity; once again using the calendar defined in *General>Settings>Work Times*.

The 'End Date' is then evaluated and displayed on the generated Assembly Order.

We will look at the effect of this when we create an Assembly Order in section 6. For now enter

the following information

Leadtime: 4 Days
Setup Duration: 6 Hrs
Run Duration: 1 Hour per 10

7.3 BOM Costs

The Standard Cost of an Item can either be manually entered against the Item via **Pricing>Item Pricing** or can be determined from the multi-level Bill Of Material structure as follows.

In describing the process let us assume the product structures defined above. I.e.

Item C uses	Item	D	2 off	Run
	Descriptor	General Time	1.0 Hrs	Run
Item A uses	Item	B	1 off	Run
	Item	C	2 off	Run
	Descriptor	General Time	1.5 Hrs	Run
	Descriptor	General Time	0.1 Hrs	Run
	Descriptor	General Time	4.0 Hrs	Setup

We should first identify the Costs against the Purchased Items and Descriptor Codes. There are numerous ways that the Costs can be established and maintained and they are covered under a separate document. For now we will manually add a Cost to the relevant Items.

Costs are required against Items B and D so go into **Pricing>Item Costing** and add and save the following into the '**Standard Cost**' field against each record.

Item B	\$10
Item D	\$4

A Cost is also required against Descriptor **GENERAL TIME**. Therefore go into **Pricing>Descriptor Costing** then add and save a Standard Cost of **\$30** per hour.

If we now get Ostendo to calculate Costs it will commence with the Item at the lowest level in a BOM and calculate the Costs at each higher level until all levels have been calculated. Once again, using the above example we would expect the following

Cost of D	=	\$4	
Cost of B	=	\$10	
Cost of C	=	\$38 calculated from	
Item		D (2 off at \$4 each)	= \$ 8
Descriptor		GENERAL TIME (1 Hour at \$30)	= \$30
		Total	-----
			\$38
Cost of A	=	\$137 calculated from	
Item		B (1 off at \$10 each)	= \$10
Item		C (2 off at \$38 each)	= \$76
Descriptor		GENERAL TIME (1.5 Hour at \$30)	= \$45
Descriptor		GENERAL TIME (0.1 Hour at \$30)	= \$ 3
Descriptor		GENERAL TIME (4 Hour at \$30 / 20)**	= \$ 6
		Total	-----
			\$140

** This is a Setup activity and the Cost per Item is the total setup cost divided by the **Quantity Use for Cost Roll-Up** quantity that you entered (above) in the 'Detail' tab

To see if this is correct go into **Assembly>Standard Cost Rollup** and click on the 'OK' Button. If you go to Item **A** you will see that the system calculated value is displayed in 'Rolled Up Cost' field. It is also shown in field 'New Std Cost' where it can be amended if required. It is the value in this field that will be used to update the Standard Cost of this Item.

Having agreed with the Calculated/Amended Standard Cost 'check' the checkbox to the left of the line and click the 'Update Standard Costs for selected Lines' button.

Go into **Pricing>Item Costing** and select the Item that you have just updated and click on the 'Detail' tab. You will see that the record's **Standard Cost** field has been updated and a history record created

7.4 By Products and Co Products

Firstly let us define what they are. **Co-Products** are Items that are ALWAYS produced whenever the Parent Item is produced whereas a **By-Product** MAYBE (but not always) produced.

1. Co-Products

To add a Co-Product to a Bill Of Material you should first create the Co-Product. Therefore go into **Inventory>Items** and add

<u>Item Code</u>	<u>Description</u>	<u>UOM</u>
Co	CoProduct	Each

Now go into **Assembly>Bills Of Material** and select Assembly 'A'. Go into the 'Detail' tab and you will see - in the lower half of the screen - a tab named 'ByProducts & CoProducts'. Go into this screen and 'Add' the following:

Code: **CO**
 Quantity: **2**
 Cost%: **40**
 Style: Select '**CoProduct**' from the drop-down list

Let's look at what you entered.

1. The **quantity** relates to the number of Co-Products that will be produced when the '**BOM Batch Size**' of the Main product is made
2. The cost of both the Main Product and the Co-Product(s) are derived from the same Components in the Bill Of Material. The **Cost%** defines what percentage of the total BOM cost will be allocated to this Co-Product. This means that the difference between the sum of all Co-products' costs and the Assembly Cost represents the Cost of the Main Product.

2. By-Products

To add a **By-Product** to a Bill Of Material you should first create the By-Product. Therefore go into **Inventory>Items** and add

<u>Item Code</u>	<u>Description</u>	<u>UOM</u>
By	ByProduct	Each

Now go into **Assembly>Bills Of Material** and select Assembly 'A'. Go into the 'Detail' tab and you will see - in the lower half of the screen - a tab named **ByProducts & CoProducts**. Go into this screen and **Add** the following:

Code: BY
Quantity: 1
Cost%: 2
Style: Select **ByProduct** from the drop-down list

Let's look at what you entered.

1. The **quantity** relates to the nominal quantity of By-Products that may be produced when the '**BOM Batch Size**' of the Main product is made
2. Unlike the Co-Product the percentage Cost contribution is NOT deducted from the receipt cost of the Main Product.

3. Affect of Co-Products and By-Products on Standard Costing

If you now run the Standard Cost Roll-up as described in Section 4 you will see that the Standard Cost of Parent Item 'A' is now **60%** of the total BOM cost. You should note that:

- Co-Product Standard Costs are not evaluated because the Co-Products themselves can be associated with numerous Main Products and therefore can have widely varying Costs.
- By-Product Costs are never evaluated during Cost Rollup as they are only evaluated during the receipt process in an Assembly Order

We will now create an Assembly Order for the above BOM, issue the Components, then receive the Main Product, Co-Product and By-Product into stock

7.5 Assembly Orders

Assembly Orders can originate from the following sources:

- **On the Fly:** An order can be raised for any Inventoried Item by directly going into the Assembly Order screen and creating the Order.
- **Source On Demand:** Any Item flagged as being identified as an Assembly and whose supply method is '**Source on Demand**' will automatically have an Order Request created. Generation of these Orders is covered under the '**Inventory Control**' training guide
- **Planned Order:** A Planned Order is a Suggested Order generated via the Replenishment Run. Generation of these Orders is covered under the '**Inventory Control**' training guide

1. Creating an Assembly Order

Go into **Assembly>Assembly Orders** and click on the **Add** button. Amend the Required Date to (say) **5** days into the future, select Item 'A' and enter an Order Quantity of **100** then click on the **Create** button

Let's have a look at what has been created:

In the '**Detail**' Tab you will see:

The '**Start Date**' is the '**Required Date**' less **4** days
 The Setup duration is **4** Hrs as held against the BOM
 The Run Duration is **10** Hrs (Order Qty **100** divided by **10** per Hour run rate)

In the 'Lines' Tab the generated components include:

From the BOM for 'A'

10	Item Code	B	100
10	Descriptor Code	GENERAL TIME	150
20	Descriptor Code	GENERAL TIME	10
20	Descriptor Code	GENERAL TIME	4

'Blown through' from the BOM for **Phantom** 'C' taking into account the Usage of 'C' in 'A'

10	Item Code	D	400
20	Descriptor Code	GENERAL TIME	200

2. Issuing Components

Assembly Order Lines can be issued in one of two ways

Direct Entry: The 'Actual' usage can be entered directly against individual Assembly Order Lines by going into the Assembly Order that you created and selecting the **'Lines'** tab.

Highlight the first Component (Item **'B'**) and then click on the **'Actual Issues'** tab. Click the **'Add'** button to create a new line in the lower part of the screen. In that record enter (say) **70** in the **'Qty'** field then click the **'Save'** button. On the **'Line Info'** bar directly above **'Qty'** field you will see a **'Stock'** button showing the current Stock and the Available Stock for this Item. If you click on this button then you will see more details about Item **'B'**

Batch Entry: A separate Assembly Issue function allows you to select lines to be issued. This can cover:

- A Range of Assembly Orders
- A Specific Order
- A Step within a specific Assembly Order

To use this option you should minimise or close out of the Assembly Order screen then go to **Inventory>Assembly Issues**. Click on the **'Add'** button to create a new batch. **'Save'** the Batch and click on the **'Lines'** tab. Click on the **'Select Assembly for Prefilling Issues'** button and select the above Assembly Order. Click the **'Create'** button to generate the Issue Lines. These will be prefilled with the outstanding Quantities to Issue. You can amend these quantities if required in addition to amending the Warehouse/Location details.

Upon completion go back to the **'Detail'** tab and click the **'Post All Issue Entries'** button.

Points to Note:

- In **Inventory>Settings>Item Rules** if field **'Disable Negative Stock for all Items'** is 'checked' then you will be denied the Issue if it causes any Item to go into negative stock
- If the above field is not 'checked' then this action will also be denied if the specific Item has been flagged as not allowing negative stock. (See **Inventory>Items** then the **'Additional Inventory Settings'** button on the **'Detail'** view)

If you now go to **Inventory>Inventory Availability** you will see that the issue has been recorded in Transaction History and current stock

3. Receipt into Stock

Assembly Orders can be received in one of two ways

Direct Entry: The 'Actual' receipt can be entered directly against the Assembly Order by going into the Assembly Order that you created above and selecting the **'Order Receipts'** tab in the **'Detail'** view.

Click the **'Add'** button to create a new receipt line in the lower part of the screen. You will see that

it is populated with the Main product ('A'). If you click on the drop-down you can select the Main Product, Co-Product, or By-Product.

Batch Entry: A separate Assembly Receipt function allows you to select an Assembly Order and prefill the receipt lines with the Main Item, Co-Products, and By-Products

To use this option you should minimise or close out of the Assembly Order screen then go to **Assembly>Assembly Receipts**. Click on the 'Add' button to create a new batch. 'Save' the Batch and click on the 'Lines' tab. Click on the 'Select Assembly for Prefilling Receipts' button and select the above Assembly Order. Click the 'Create' button to generate the Receipt Lines. These will be prefilled with the outstanding Main Item and Co-Product Quantities to Receive. You can include By-Products by clicking on the 'Add' Button and adding these to the list. Amend the displayed quantities if required in addition to amending the Warehouse/Location details. You should also note that if the product has 'sub-level' variations (Colour, Size, Batch, etc) then these should be specified in this screen prior to posting the receipts.

Upon completion go back to the 'Detail' tab and click the 'Post All Receipt Entries' button.

If you now go to **Inventory>Inventory Availability** you will see that the receipt has been recorded in Transaction History and current stock

7.6 Labour

The above process has concentrated on Items and Descriptors being used to make the end product and can include Labour activities covered by a 'Labour' style Descriptor.

If, however, you are going to record **Employee** times against a Labour Activity then you should not use a 'Labour' style Descriptor but use a 'Labour Code' instead.

The Labour Code identifies the skill level and hence the 'Planned Cost' of an activity whereas the Employee books time against the Labour Code to record the 'Actual Cost' of carrying out that activity.

1. Adding a Labour Code to a BOM

Go into **Labour>Labour Codes** and select **LAB-ASSEMBLY**. You will see that it has a Standard Cost (\$26) linked to it. This Cost will be used to determine the 'Planned' Cost of an Assembly Order.

Go into **Assembly>Bills of Material** and select Assembly **C** that you created in section 3.2.1. Click on the 'Lines' tab at the top of the screen and add the following Labour Code to the existing Lines

<u>Line</u>	<u>Line Type</u>	<u>Code</u>	<u>Qty</u>
30	Labour Code	LAB-ASSEMBLY	3

Finally, go into **Assembly>Standard Cost Rollup** and carry out a 'Cost Roll Up'. If you select Item **C** and then click on the 'Detail' tab you will see the Cost for **LAB-ASSEMBLY** comes from the above record.

2. Booking Employee Times

If you go into **Labour>Employees** and click on 'Bob Drum' you will see - in the 'Detail' tab - that an 'Employee Unit Cost' has been entered. If you click on the 'Labour Codes' tab you will see an entry that signifies that when **Bob Drum** carries out work against a '**LAB-ASSEMBLY**' then Bob

Drum's Unit Cost will be used PLUS a Fixed and Variable Overhead Cost. You will see that the total Cost per hour comes to **\$37**

Go into **Assembly Orders** and create a new Assembly Order for quantity **1** of Item **C**. In the Assembly Order **Details** screen you will see that the Planned Order Costs are **\$121** comprising of

Item Code	D	2 @ \$4	= \$8	
Descriptor Code	GENERAL TIME	1 @ \$35	= \$35	
Labour Code	LAB-ASSEMBLY	3 @ \$26	= \$78	

Go into the **'Lines'** tab on the Assembly Order and click on the **'Actual Issues'** tab in the lower part of the screen.

Issue the required quantities against both the Item Code and Descriptor Code.

Against the Labour Code Line select Employee **'Bob Drum'** and enter **3 Hrs**

If you now go to the Assembly Order **Details** screen you will see that the Actual Order Costs are **\$154** comprising of

Item Code	D	2 @ \$4	= \$8	
Descriptor Code	GENERAL TIME	1 @ \$35	= \$35	
Labour Code	LAB-ASSEMBLY	3 @ \$37	= \$111	

In this instance the recovery Cost of **\$37** per hour is taken from the Employee record when that Employee books time against the Labour Code.

A detailed Cost summary sheet can be printed by clicking on the **'Reports'** Button to the right of the screen and selecting Report **'Assembly Costing Sheet'**

7.7 Assembly Order Backflushing

Backflushing is used during Assembly Order receipts to automatically issue components required to make the Assembled Item at the time the Item is received into stock.

1. Preparation

You can define if you want all Assembly receipts to automatically backflush or only have this feature against selected Items.

To set the Backflush to be all receipts go to **Inventory>Settings>Item Rules** and select **'All Items'** under drop-down against field **'Assembly Backflush Policy'**.

In this training session we will backflush against a specific Item. Therefore:

- Go into **Inventory>Settings>Item Rules** and select **'Item Specific'** under drop-down against field **'Assembly Backflush Policy'**.
- To identify the specific Item go into **Inventory>Items** and select Item **1105-2184** (Handle Assembly). In the **'Detail'** screen click on the **'Additional Inventory Settings'** button. In that panel 'check' the **'Backflush Issues on Assembly Receipts'** checkbox

2. Backflushing in action

2.1. Order Creation

Go into **Assembly>Assembly Orders** and create an Assembly Order for **40** off **1105-2184**

If you click on the 'Lines' tab within the Order you will see that the BOM was copied with the following amounts

Item Code **760-2176** (Tube-Stainless Steel-25mm1200mm) **40** off
Labour Code **LAB-ASSEMBLY** (Assembly Labour) **40** hours
Item Code **900-2182** (Handle Grip-Rubber-25mm) **80** off

Add the following line to the Order Line

Descriptor Code **GENERALTIME** (Time - Non Employee Related) **10** hrs

Note the Order Number then close out of the Assembly Order screen

2.2. Receive and Backflush

Go to **Assembly>Assembly Receipts** and create a Receipt batch. Click on the '**Lines**' tab and - in that screen - click on the '**Select Assembly for prefilling receipts**' button. On the displayed panel select the above Assembly Order. This should create a single line displaying Item **1105-2184**. Scroll to the '**Qty**' field and change the prefilled quantity to **10**. Save the record then click on the '**Detail**' tab. On the Detail screen click on the '**Post all Receipt Entries**' button. Then close out of the Assembly receipts screen.

Go back to **Assembly>Assembly Orders** and select the above Order and click on the '**Lines**' tab. You will see that it has backflushed the Item Codes and the Descriptor but NOT the Labour Code. The reason for this is that Employee Times booked through the Timesheet (**Labour>Timesheets**) will be used to update Labour Activities. Having said this you may not wish to use the Timesheet function but simply get Ostendo to 'Backflush' Labour Code activities. We will do this in the next step.

2.3. Backflush Labour Codes

Go into **Assembly>Settings>Assembly Rules** and 'check' the checkbox '**Include Labour When Backflushing**'. Once 'checked' you should also nominate a default Employee (say **Bob Drum**) from the drop-down against '**Default Employee for Backflushing**'.

Now go to **Assembly>Assembly Receipts** and create a Receipt batch. Click on the '**Lines**' tab and - in that screen - click on the '**Select Assembly for prefilling receipts**' button. On the displayed panel select the above Assembly Order. As before scroll to the '**Qty**' field and change the prefilled quantity to **10**. Save the record then click on the '**Detail**' tab. On the Detail screen click on the '**Post all Receipt Entries**' button. Then close out of the Assembly receipts screen.

Go back to **Assembly>Assembly Orders** and select the above Order and click on the '**Lines**' tab. You will see that it has backflushed the Labour Code in addition to the Item and Descriptor Codes. You may wish to click on the '**Actual Issues**' tab in the lower part of this screen to look at the Issue History for the selected line.

2.4. Items with variants

If the Item has sub-level variants (such as Colour, Serial Number, Batch, etc) then the backflush cannot automatically take place. In this instance an 'Issue Batch' is generated against which you should manually state which variant has been issued. Let's see how this works:

Go back to **Assembly>Assembly Orders** and select the above Order and click on the '**Lines**' tab. Add a new line containing the following Item that has 3 variations of Colour

Item Code **OC-7451** (Office Chair - Fabric Executive) **40** off

Go back into **Assembly>Assembly Receipts** and create another Receipt batch. Click on the 'Lines' tab and - in that screen - click on the '**Select Assembly for prefilling receipts**' button. On the displayed panel select the above Assembly Order. As before scroll to the 'Qty' field and change the prefilled quantity to **10**. Save the record then click on the 'Detail' tab. On the Detail screen click on the '**Post all Receipt Entries**' button. Then close out of the Assembly receipts screen.

If you go back to **Assembly>Assembly Orders** and select the above Order and click on the 'Lines' tab. You will see that Backflushing has not been carried out against **OC-7451**. However, if you go to **Assembly>Assembly Issues** you will see that an Issue Batch has been created to cover this. That Issue Batch will require the specific variant to be selected before it is issued

2.5. Source On Demand

If the Assembly Order Component Line is an Item Code with 'Supply Method' of '**Source on Demand**' then the Item will not be backflushed because a supply Order is scheduled to be generated to satisfy this demand

Similarly, if the Assembly Order Component Line is a Descriptor with 'Source By' of '**Purchasing**' then the same condition will apply

7.8 Assembly Order Closure

You can change an Assembly Order status to 'Closed' by either going into the Order 'Detail' screen and changing the status or going into screen **Assembly>Assembly Closing**

Once the Assembly Order status is changed to '**Closed**' any remaining Work In Progress costs are posted to an '**Assembly Order Variance**' Account. Subsequent bookings are allowed against this order but any costs will by-pass the WIP Control Account and post directly to '**Assembly Order Variance**'.

7.9 Custom Products

Ostendo has two versions of Custom Products.

Manual: A simple selection of components and quantities from a pre-defined Bill of Material

Rules Based: A more complex Custom Product configurator with inbuilt conditional inclusions and exclusions, calculations, and other user-specific Rules

1. Custom Products - Manual

To demonstrate this feature we should first set up the Product and its component variables

1.1. Defining the Custom Product

Go to **Inventory>Items** and create a new Item (say) '**Wagon**' with the following fields completed:

Unit: Each
Description: Configured Wagon
Default Supply Method: Source on Demand
Sourced By: Custom
Configured By: Manual
Batch No: 'checked'

'Save' the record

Go to **Assembly>Bills Of Material** and click the 'Add' button. On the presented panel:

Select 'Wagon' from the drop-down under field 'Item Code'

Select the 'Copy Bill of Material from another Item' Radio Button

Select 'WAGON-2189' from the drop down in the field below the Radio Button

Click the 'Create' button

On the generated BOM go into the 'Lines' tab. We are going to provide the option to have Green, Red, Blue, or Yellow paint. Therefore add the following lines to the BOM

Line No: 150
Item Code: 1500-2186
Qty: 1.5
Step: Painting

Line No: 160
Item Code: 1500-2187
Qty: 1.5
Step: Painting

Line No: 170
Item Code: 1500-2188
Qty: 1.5
Step: Painting

1.2. Generating the Custom Product

Go to **Sales>Sales Orders** and click the 'Add' button. On the presented panel select Customer 'Jim Gold & Co Ltd' and then click the 'Create Order' button.

Click on the 'Lines' tab in the created order and then click the 'Add' button. The cursor will be positioned in the 'Code' field. Click on the 'Spyglass' icon to the right of that field and select 'Wagon'. You will notice that a new icon has appeared in the 'Code' field. Click on this and a panel similar to the above BOM screen will appear. You can click on the 'Lines' tab and adjust the required quantities to suit the configuration. Hint: you can drag a column heading left or right and drop at the desired viewing position. You also have the option to delete lines or add new lines and make it specific to this configuration.

If you click onto the **Costing and Pricing** tab you will see the current costs of the selections that you've made and the anticipated Sell Price. Of course you can amend the Markup% and click the 'Save' button to adjust the Sell Price. You can click on the 'Update Sales Line Price' to copy this revised price to the Sales Order. Close out of this screen when done

You can revise the configuration at any time up to when you actually generate an Assembly Order from this BOM.

1.3. Generating the Assembly Order

From within the Sales Order click on the 'Related' button to the left of the screen and select 'Create Required Assembly Orders'. On the presented screen 'check' the 'Wagon' and then click on the 'Generate Orders for Selected requirements' button situated at the bottom-right of the screen.

Still within the Sales Order click on the 'Line Source' in the lower part of the screen where you will

see the generated Assembly Order number.

If you go to **Assembly>Assembly Orders** you will see the generated Assembly Order for the Wagon. This order contains the information specifically configured in the Sales Order.

2. Custom Products – Rules Based

It is not the objective here to carry out training in Custom Scripting. Prior knowledge of Pascal scripting is desirable though not essential. In this exercise we will look at an existing script in your database and use that to go through the Configuration process

2.1. Defining the Custom Product

Go to **Inventory>Items** and select Item '**OD-7001**' (Office Desk Executive (Configurable)). You will notice the following settings that define this as a Custom Manufactured Product

Default Supply Method: Source on Demand
Sourced By: Custom
Configured By: Rules
Batch No: 'checked'

Now go to **Assembly>Custom Products** and, select product **OD-7001** and click on the '**Script**' tab. The script can contain three main sections:

Variables: defining what variables are to be evaluated
Questions: Defining the questions to be asked
Actions: Defining the required actions for each answer given

2.2. Generating the Custom Product

Go to **Sales>Sales Orders** and select the Order you created in 10.1.2. Click on the '**Lines**' tab in the order and then click the '**Add**' button. The cursor will be positioned in the '**Code**' field. Click on the 'Spyglass' icon to the right of that field and select '**OD-7001**'. A panel will be presented that shows the first question identified in the Script. If you answer this question then – based on the rules as a result of the first answer – another question will be presented.

Continue through the questions, answering as you go along. After confirming that the configuration is complete Ostendo will calculate the cost and arrive at a Sell Price in the Sales order Line. You can, of course, amend the Sell price is required.

Behind the scenes Ostendo has also created a Bill of Material for this specific configuration and that will be used to generate the Assembly Order.

2.3. Generating the Assembly Order

From within the Sales Order click on the '**Related**' button to the left of the screen and select '**Create Required Assembly Orders**'. On the presented screen 'check' the '**OD-7001**' and then click on the '**Generate Orders for Selected requirements**' button situated at the bottom-right of the screen.

Still within the Sales Order click on the '**Line Source**' in the lower part of the screen where you will see the generated Assembly Order number.

If you go to **Assembly>Assembly Orders** you will see the generated Assembly Order for the Office Desk. This order contains the information specifically configured in the Sales Order.

8 7. Job Orders

Job Order coverage can range from small Jobbing Orders through to large 'Project' type activities. These can contain the following elements:

- Single or Multiple Task Related activities
- Resource (Labour, Heavy Plant, Machines, etc) Scheduling
- Visual Planning Board
- Ongoing Cost Accounting with full Job/Project Cost visibility
- Cost and Timeline projections
- Billing schedules and Retentions
- Multiple Jobs linked to single Project

This series of exercises will start off with the available invoicing options then proceed with a simple Job and progress through the various levels of complexity. We will conclude coverage of Job Orders by addressing the Job Calendar

8.1 Customers

If you have not been through the Customer Creation and Maintenance process (normally carried out in the Sales Training Exercises) then you should commence with this section, otherwise go to Section 3

1. Preparation

User-Maintained Tables: The following tables are used when creating Customer records. Take a look at them. There are some defaults already set up but you may wish to add more or amend the current records:

Mandatory Tables

The following three fields are mandatory and validated against a separate table when creating a Customer record. Within each Table, however, you can nominate a 'default' that will populate a Customer record when adding a new Customer record.

Customer Types: Segregates Customers into logical groups (e.g. Trade, Retail). You can maintain these via **Sales>Settings>Customer Types**

Tax Group: To facilitate Customer/Item Tax code evaluation. You can maintain these via **File>Financial Configuration>Credit Terms**

Terms: Days from (Invoice, EOM, End of Next Month) plus Early Payment Discount. You can maintain these via **File>Financial Configuration>Tax Groups**

Optional Tables

The following fields are optional and, when used, are validated against a separate table when creating or maintaining a Customer record.

Sales Regions: For Sales Analysis purposes. You can maintain Sales Regions via **Sales>Settings>Sales Regions**

Customer Codes: A Customer Code can be used to 'group' Customers within the same group (Example: Retail, Trade, etc). To maintain Customer Codes go into **Sales>Settings>Customer Codes**

Shipping Methods: This allows you to pre-define that Shipping Method applicable to the

Customer and if the method is Taxable (along with the Tax Rate). To maintain Shipping Methods go into **Sales>Setting>Customer Shipping Methods**

Sales Person: You can allocate a default Salesperson to a Customer. This can be amended at Sales Order level if required. To create a Salesperson go into **Labour>Employees** and create an Employee. In the main Employee screen you can identify the Employee as being a Salesperson

Lead Source: This can be used for Sales Analysis purposes. To maintain Lead Sources go into **Sales>Settings>Customer Lead Sources**

Price Level: This is used in the Pricing function to determine the sell price of an Item or Descriptor based upon the Pricing Level held against this Customer. Examples of Pricing Levels are Retail, Trade, etc. To maintain Price Levels go into **Pricing>Settings>Price Levels**. The use of Price Levels will be covered in more detail later in this document

Rate Level: This is used in the Pricing function to determine the Charge Rate for a Labour activity based upon the Pricing Level held against this Customer. Examples of Rate Levels are Retail, Trade, etc. To maintain Rate Levels go into **Pricing>Settings>Rate Levels**. The use of Rate Levels will be covered in more detail later in this document

Invoicing Group: This is (optionally) used when generating and printing Invoices where you can restrict the batch selection to Customers within a specific Invoicing Group. To maintain Invoicing Groups go to **Sales>Settings>Invoicing Groups**.

Statement Cycle Code: Enables Statements to be printed relative to their 'cycle'. To maintain Statement Cycle Codes go to **Sales>Settings>Statement Cycles**

2. Create Customer records

Go into **Sales>Customers** and add a new Customer record. You will see that the created record is prefilled with the defaults identified in the previous section. These fields can be amended if required.

Having created the Customer record the following linked records can be created and maintained

3. Customer Additional Fields

There are two levels where Additional fields could be required against Customers:

- Global Fields that apply to ALL Customers (Example:- '**Valued Customer**' flag, etc)
- Properties that apply to some Customers (Example:- **Overseas Country**)

3.1. Global Additional Fields

Go into **File>System Configuration>Additional Fields** and click on the '**Add**' button. On the displayed line enter the following:

Module: Select '**Customers**'

Caption: Enter the Additional Field name (Example: Valued Customer)

Field Type: From the drop-down list select the format of the field. The options are:

- **Text:** Any data format can be entered in a Text field
 - **Decimal:** Allows entry of numbers and decimals
 - **Integer:** Allows entry of whole numbers only
 - **Currency:** Shows Currency symbol and decimals as defined in Regional Settings
 - **Yes/No:** Shows a checkbox which can be checked/unchecked
 - **Date:** Contains a drop-down calendar for selection of a date
-

- **Time:** Displays format HH:MM:SS for entry of a time of day
- Value List:** This allows you to define any specific entries to which a drop-down list - during data entry - is restricted

'Save' the entry and 'Close' the screen when done

If you now go to the Customer screen (**Sales>Customers**) and click on the detail tab you will see a 'tab' (**Additional Fields**) in the centre-left of the screen. Click on this tab and enter some data into this field then 'Save' the record.

You can view these additional fields in the Customer 'List' view if required by going into the Customer's List screen and 'right mouse' in the centre panel. Select 'Customize List Fields' from the displayed panel. (Note: If that option is not visible then go to **File>System Configuration>User Security and Options** and go to the 'User Options' tab for the current User. 'Check' both the 'Save Grid Layouts' and 'List Customising' checkboxes.) On the displayed panel 'check' **Additional Field_1** and give it a 'Display Name' of (say) 'Valued Customer'. 'Save' the entry. The field will now display on the List screen where you can sort and filter as necessary.

4. Customer Contacts

You can have multiple Contacts against each Customer. To create these contacts click on the 'Related' button and select 'Contacts'. Alternatively you can go to **CRM>Contacts** and create the contacts via that screen using 'Contact Type' = **Customer** and selecting this Customer under 'Company Name'.

5. Delivery Addresses

You can have multiple Delivery Addresses against a Customer. Any one of these can be selected during Order entry and made specific to the Order. To create additional Addresses click on the 'Additional Physical Addresses' in the main Customer screen

6. Customer Properties

This feature allows you to define a 'Property' (Example: Corporate Status) and then link that property to selected Customers with a value that is specific to each Customer (Example: Gold, Silver, Bronze)

To demonstrate this, go into **General>Settings>General Properties** and select 'Customer' from the drop-down under 'Module'. Now add 'Corporate Status' with Property Type of 'Text' and the following entries - on separate lines - in the Property Values field (**Gold**, **Silver**, and **Bronze**). Click on 'Save' and then 'Close'

Now go to **Sales>Customers** and select the customer you created above. Click on the 'Related' Button on the right of the screen and select 'Customer Properties'. Click the 'Add' button and:

- Add a line using property 'Corporate Status'
- Select 'Gold' from the drop-down under column 'Value'
- 'Check' the 'Copy to Job Lines' checkbox

7. Customer Images

You can add multiple images (pictures, drawings, maps, plans, etc) to a Customer record. These can be printed on all documents where the Customer is used. Go to **Sales>Customers** and select the Customer that you created above. Click on the 'Related' Button on the right of the screen and select 'Customer Images'. Click the 'Add/Edit' button and:

- Give the Image a short Name
- Point the program to where the image is located on your computer network
- 'Check' the '**Copy to Jobs**' checkbox then '**Save**' and exit the screen

2.8. Customer Documents

You can add multiple documents to a Customer record. These can be printed along with all documents where the Customer is used. Go to **Sales>Customers** and select the Customer that you created above. Click on the '**Related**' Button on the right of the screen and select '**Customer Documents**'. Click the '**Add/Edit**' button and:

- Give the Document a short Name
- Point the program to where the document is located on your computer network
- 'Check' the '**Copy to Jobs**' box then save and exit the screen

8. History Notes

This function allows you to link multiple time-stamped notes to a Customer record. Against selected History Notes you can also add a dated reminder so that Ostendo will prompt you of the reminder once the date is reached. Go to **Sales>Customers** and select the Customer that you created above. Click on the '**Related**' Button on the right of the screen and select '**Customer History Notes**'. Click the '**Add**' button and:

- Enter some history notes
- 'check' the '**Follow-Up**' required checkbox and select a date from the adjacent drop-down calendar. '**Save**' the History Note and exit the screen
- To see the '**Follow Up**' in action you should first change the company by clicking on **File>Change Company** and selecting **DEMO** then sign in as **ADMIN/pass**. Repeat this and go back to company '**Training**'. Upon sign-in as **ADMIN/pass** the alert should present itself if the Follow-Up date is current
- Note: In the CRM Module if a 'Call' was raised and subsequently 'closed' against a Customer then any activity notes entered against that Call will be posted to the Customer's History file

8.2 Preparation

You may wish to address the following areas before creating a Job Order

Auto Numbering: You can nominate numbers from which the program auto generates an incremental number for Job Order identity. Go into **File>System Configuration>System Settings** then click on the '**System Numbering**' tab. The references under the '**Jobs**' Module will be addressed in this exercise in addition to the Sales Invoice

Job Types: Whenever a Job is created it is given a Job Type. This allows Ostendo to focus on what the Order applies to and how it is to be invoiced. If you go to **Jobs>Settings>Job Types** you will see that some Job Types have already been set up. These cover:

Job Style defines if the Order covers

- Customer Order (for Contracting, Jobbing, Consulting, etc)
- Customer Asset (normally for Warranty and/or Servicing)
- Company Asset (for Plant Maintenance or "In-house work)
- Rental (Equipment or Service Hire by Customer)

Invoice Style relates to how this Job is to be Invoiced. The options are:

Scheduled: Invoice(s) can be pre-defined in an Invoicing Schedule. The total value of the Schedule is based on either:

- The sum of the sell prices of the included Job Lines, or
- A Fixed Price

Actual: Based on Actual Usage incurred. This usage may be charged at Sell

Price or Cost-Plus.

No Invoice: No Invoice required (Example: Plant Maintenance)

Job Rules: This allows you to define rules that apply to Job Orders. Go into **Jobs>Settings>Job Rules** and adjust the rules to suit your requirements. You should refer to the Help (**Help>Reference**) then select **Jobs>Settings>Job Rules** for assistance in what rules to apply.

Job Categories: User-defined categories that allow you to segregate Jobs for analysis purposes. For example:- Public, Council, Government, etc Go into **Jobs>Settings>Job Categories** and add couple of Categories

Tracking Codes: Enables lower-level tracking to main Job Status. This will be used when we go into the Job Calendar. For now go into **Jobs>Settings>Job Tracking** and add the following:

Tracking Code: **Scheduled** Description: **Service Engineer Scheduled**

Tracking Code: **On Site** Description: **Service Engineer On-Site**

Allocate a tracking colour to each Tracking Code

Lists: Lists are simply a User-defined List of Items, Descriptors, and Labour Codes that can be called up in Job Lines to facilitate 'batch' input when adding lines to the Job. Go into **Inventory>Lists** and you will a couple of pre-defined lists. We will reference these Lists during the course of our exercises

8.3 Functions used in Quotations and Job Orders

Go into **Jobs>Job Orders** and click the **'Add'** button. A panel will appear. Select Job Type **'Progress'**. (Note: This Job Type is maintained as described in 3. - above). Select Customer **'Jim Gold & Co Ltd'** then click the **'Create Order'** button

We will go into the fields in the Order Header in more detail later. For now click on the **'Lines'** tab to enter Job Order Lines

1. Adding and Maintaining Job Lines

Many options are available for adding lines to the Sales Quote or Order such as:

- Selecting from a user-defined List
- Selecting Items in batch
- Selecting Descriptors in batch
- Selecting multiple Lines from a Supplier Catalogue
- Selecting a single Item
- Selecting a single Descriptor
- Selecting a single Line from a Supplier Catalogue
- Selecting a complete Kitset of Items/Descriptors
- Configuring a 'Custom Product'

1.1. Selecting from a user-defined List

Click on the **'List'** button on the **'Batch Entry'** bar that is situated across the upper part of the screen. A panel is presented that displays all pre-defined Lists. These are user-defined Lists that enable you to make multiple selections and copy those selections to the Order. You will see that 2 Lists already exist in the database. You can create your own Lists via **Inventory>Lists**. Upon selecting a **'List'** a further panel will appear showing all lines in the List (Items, Descriptors, Labour). Select a couple of lines from the List and click the **'Create Lines from Selected Contents'** button. Each selected line will become a Sales Order Line in its own right.

1.2. Selecting Items in batch

Click on the **'Items'** button on the **'Batch Entry'** bar. A panel is presented that displays all Items (excluding those with status **'Obsolete'**) in Ostendo. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the **'Add to Selected'** button

In the lower panel you can now amend the required quantity.

Once the full Item selection has been made then click the **'Create Lines from selected contents'** button to add the lines to the Job Order

1.3. Selecting Descriptors in batch

Click on the **'Descriptors'** button on the **'Batch Entry'** bar. A panel is presented that displays all **'Active'** Descriptors that are designated 'for general purpose use'. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the **'Add to Selected'** button

In the lower panel you can now amend the required quantity.

Once the full Descriptor selection has been made then click the **'Create Lines from selected contents'** button to add the lines to the Job Order

1.4. Selecting multiple lines from a Supplier Catalogue

Click on the **'Catalogue Items'** button on the **'Batch Entry'** bar. A panel is presented that shows all Supplier Catalogues; select the Catalogue from which Items are to be extracted. (Note: If you have only one Supplier Catalogue then this (Catalogue Selection) step is ignored by Ostendo).

All Items in the selected Supplier Catalogue are now displayed. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the **'Add to Selected'** button

In the lower panel you can now amend the required quantity.

Once the full selection has been made from the Catalogue then click the **'Create Lines from selected contents'** button to add the lines to the Job Order

1.5. Selecting Templates

Templates are described in more detail later in this 'Job Orders' series of exercises. We will come back to this at that time.

1.6. Selecting a single Item

This can be used as an alternative to the above 'Batch' selection. Click on the **'Add'** button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against **'Line Type'** select **'Item Code'** then go to the next field (**'Code'**) to select the specific Item from the drop-down list. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select any Item.

1.7. Selecting a single Descriptor

This can be used as an alternative to the above 'Batch' selection. Click on the **'Add'** button

located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against 'Line Type' select 'Descriptor Code' then go to the next field ('Code') to select the specific Descriptor from the drop-down list. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select any Descriptor

1.8. Selecting a single Line from a Supplier Catalogue

This can be used as an alternative to the above 'Batch' selection. Click on the 'Add' button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against 'Line Type' select 'Catalogue Code' then go to the next field ('Code') to select the select the Catalogue from which Items are to be extracted. (Note: If you have only one Supplier Catalogue then this step is ignored by Ostendo). Having selected the Catalogue then all lines within that catalogue are displayed. Select the specific Catalogue Item that you want to add to the Job Order Line. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select Item **PT-3220** from the 'Power Tools' Catalogue

1.9. Selecting from a complete Kitset of Items/Descriptors

A Kitset is a pre-defined 'kit' of Items and/or Descriptors that are sold as a single combined unit. To look at current Kitsets in Ostendo go to Sales>Kitsets where you will see a single Kitset 'SPARESKIT'. If you wish to create your own Kitset then carry out the following:

- Create a Descriptor (*Inventory>Descriptors*) and give it a Descriptor Classification of 'The Descriptor is used as a Kitset Code'.
- Go into *Sales>Kitsets* and 'Add' a new Kitset by selecting the above Descriptor
- Go into the 'Lines' tab for the Kitset and add a couple of Items and Descriptors to the Kitset

Having created the Kitset (or using Kitset 'SPARESKIT') then add it to the Job Order Line by going to the Job Order Lines screen and clicking the 'Add' button located to the right of the screen. This will allow you to add details of a single Kitset to be made in the lower part of the Order Line screen. From the drop-down against 'Line Type' select 'Kitset Code' then go to the next field ('Code') to select the select the Kitset from current list of Kitsets. All the remaining information (except the 'greyed out' fields) can be amended as required.

1.10. Configuring a Custom Product

Ostendo has two versions of Custom Products.

Manual: A simple selection of components and quantities from a pre-defined Bill of Material

Rules Based: A more complex Custom Product configurator with inbuilt conditional inclusions and exclusions, calculations, and other user-specific Rules

Exercises covering these can be found in Section 10 of Training Guide 5 - Assembly Orders

2. Additional Order Line Information

Dependent upon the type of Job Order Line being addressed the following additional information is available

2.1. Order Values Bar

Just below the upper multi-line panel you should see a bar titled 'Order Values' which shows the current Planned Values in the Job Order. If you cannot see the Order Values Bar then go into

Jobs>Settings>Job Types and - for the selected Job Type - 'check' field '**Display Values Bar**'.

Similarly you can click the drop-down against '**Order Values**' and select '**Actual Values**' to display the current 'Actual' values against the Job. If you cannot see the drop-down against '**Order Values**' then go into **Jobs>Settings>Job Types** and - for the selected Job Type - 'check' field '**Show Costs**'.

2.2. Line Info Band

Just above the lower detail panel you will see a band called '**Line Info**'. The following buttons are presented in this band where applicable:

Add-On Sales: If the Item has 'Add-On Sales' then this button will allow you to ask any 'Add-On Sales' questions and take orders for those 'Add-On' Items being purchased. To see this in action we will offer an extended Warranty.

Go into **Inventory>Descriptors** and add a new Descriptor '**Extended Warranty**'. Within that screen:

- Enter a Sell Price of **\$20**
- 'Check' the '**Sales Warranty Applies**' checkbox
- Select **36M-PANDL** from the drop-down list in the adjacent field

Go into **Inventory>Items** and select **WAGON-2189**. 'Check' the '**Add On Sales Apply**' checkbox and click on the adjacent '**Item Add-On Sales**' button. In the presented panel add the '**Extended Warranty**' Descriptor.

Finally, using the Job Order created above, add '**WAGON-2189**' to the line. The '**Add-On Sales Items**' button will be displayed. If you click on this button then you can offer the Add-On Sales Items for inclusion with this Order Line

Qty Break Pricing: If an Item or Descriptor has Quantity Break Pricing then this button will be presented which, when pressed, will show the Quantity Break details. To see this in action:

- Go into the Customer record that you created in Exercise 2. On the Customer '**Detail**' screen click on the '**Pricing and Invoicing**' tab then select '**Retail**' from the drop-down list under '**Pricing Level**'.
- Go into **Pricing>Item Pricing** and select '**WAGON-2189**' then click on the '**Detail**' tab. Add a new line with Price Level '**Retail**' and apply your own Price Breaks
- Finally, go back to the above Job Order and add '**WAGON-2189**' to the order line. The Qty Break Pricing button will now appear on the '**Line Info**' band

Alternate Items: If an Item has an Alternative Item referenced to it then this button is presented so that you can view information about that alternative. To see how this works let us suppose that Item **5000-2011** (Cat 6 Network Cable - 5 Metres) could be supplied as an alternative to Item **5000-2010** (Cat 6 Network Cable - 1.2 Metres). Go into **Inventory>Items** and select Item **5000-2010**. In the Detail view click on the '**Additional Inventory Settings**' button and 'check' the '**Alternate Item Available**' checkbox. Select Item **5000-2011** from the drop-down list in the field immediately underneath the checkbox.

Finally, using the Job Order created above, add '**5000-2010**' to the line. The Alternate Item button will now appear on the '**Line Info**' band.

Stock: If any Order Line is an Ostendo Item then this button will be displayed. It shows the current **On-Hand quantity** (in stock), **Available Quantity** (On-Hand + Supply - Demand) plus the Item's basic **Unit of Measure**. Clicking on this button will show further details.

2.3. Order Line 'Tabs'

Dependent upon the type of Job Order Line the following 'Tabs' will be presented

Line Source: Based upon the 'Line Type/Supply Method/Sourced By' combination this panel shows from where the line is provisioned. The following options are covered.

- Item Code/Internal - An Assembly Order will be raised to cover this demand
- Item Code/Purchasing - A Purchase Order will cover this demand
- Descriptor/Internal - No Order planned but internal Resource is scheduled
- Descriptor/Purchasing - A Purchase Order will cover this demand
- Catalogue Item - A Purchase Order from the Catalogue Supplier will be raised

You should note that you have the option to split the Line Source record to procure the line from multiple Sources. To do this click on the 'Add' Button. This will create a new line into which you can add the new procurement source. You will also see a button 'Edit Source Quantities' just above the created line. Clicking on this button brings up a separate panel for re-allocation of Order Line's quantities.

If you now select the Catalogue Item in the upper panel you will see that this line has a source linked to the Catalogue Supplier. If you click on the 'Related' Button to the right of the screen then select 'Create Required Purchase Orders' a screen will be presented for you to convert the Purchased 'Source On Demand' Lines into physical Purchase orders. Try converting the above Catalogue Item ('Check' the Select checkbox and click the 'Generate Orders for selected requirements' button). If you return to the 'Line Source' tab you will see the Purchase Order reference appears in the last field.

Line Properties: Any specific property values held against an Item, Descriptor, Kitset, or Catalogue Item are copied to the Job Order Line. You have the option to amend or delete current properties or even add new properties.

To see this in action you should first create the 'Property' via **General>Settings>General Properties** and add 'Voltage' with Property Type of 'Text' and the following entries - on separate lines - in the Property Values field (**115 Volts**, and **230 Volts**). Click on 'Save' and then 'Close'

Now go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the 'Related' Button on the right of the screen and select 'Item Properties'. Click the 'Add' button and:

- Add a line using the above property
- Select the Voltage from the drop-down list
- Select the specific Voltage from the drop-down under column 'Value'
- 'Check' the 'Copy to Job Lines' checkbox

The selected property and value will accompany the Item whenever it is used in a Sales Order. To demonstrate this go into the Job Order that you were using above and select the 'Lines' tab and add an Order Line for the above Item. Click on the 'Line Properties' tab and the Properties will have been copied from the Item **485-2267**. You will see the 'Line Properties' appear on the 'Line Info' bar

Planned Variants: This is visible if the line covers an Item that has 'sub-level' variations of Colour, Size, or Grade. You can define the specific variants required against this Sales Order Line by clicking on the 'Add' button and entering the variants. The total quantity of sub-level variants must equal the Line Quantity. To see this in action you should go into **Inventory>Items** and select Item **485-2268** (Internal Spotlight 250 Watt). 'Check' the 'Colour' checkbox in the 'Detail' screen. Click on the 'Colours' Button to the right of the checkbox and add colours 'Red' and 'Green'. Close out of the Item, screen and then go to the Sales Order that you were using above and select the '

Lines tab and add an Order Line for Item **485-2267**. Click on the 'Line Variants' tab and enter the quantity of each variant required in the Job Order.

Actual Issues: If you click on this tab then you can Issue the lines. You should note, however that:

- For Stock Items this requires entry of a quantity and a Warehouse/Location. This is defaulted to that held against the Item but can be amended. If the Item has 'variants' (Serial Number, Lot Number, Colour, Grade, etc) then this must also be selected from a list of current stocked variants. To look at the current list click on the 'spyglass' icon to the right of the **Qty** field
- For non-stock Items (Descriptors, Kitsets, Catalogue Items) it is simply a pick date and quantity

Kit Contents: When a Kitset is copied to the Job Order Line its content is shown here. The quantity per kit against each Line can be amended if required. The actual Issue of the Kit is carried out via the 'Picked Lines' tab. If you go into **Sales>Kitsets** you will see that a kitset has already been entered. Click on the '**Lines**' tab to view the 3 entries that comprise the Kitset. Go to the Job Order that you were using above and select the '**Lines**' tab and add an Order Line. In the drop-down under '**Line Type**' select '**Kitset Code**' then select the Kitset from the dropdown under '**Code**'. You will see that the tab '**Kitset Contents**' will now be displayed on the Line Info bar. Click on this to view the content of the Kit

Warranty: This enables you to adjust planned Warranty records that may exist against this Line or add new Warranty records as required.

To see this in action go into **Service>Warranty Definitions** and view the current Warranty Codes that are in Ostendo. Add your own as required. Now go to into **Inventory>Items** and select Item **1800-2190** (Rear Wheel Assembly). 'Check' the '**Sales Warranty Applies**' checkbox in the '**Detail**' screen. Click on the drop-down to the right of the checkbox and select one of the Warranty definitions then '**Save**' the record.

Go to the Job Order that you were using above and select the '**Lines**' tab and add an Order Line and select Item **1800-2190**. If you click on the 'Warranty' tab you will see that the Warranty definition has been copied to the Job Order Line. You can amend this, or add new Warranty records that will be specific to this Order Line.

'**Pick**' the line as described above then go to **Service>Warranty List** where you will see that a Warranty record has been generated for this Job Order.

2.4. Order Line Notes

At the bottom of the '**Lines**' screen there is space to put unlimited Notes that apply to the Order Line. If you click in the Notes area you will see two Icons appearing in the top-right of the field. If you click on the first Icon then the Notes filed will occupy a much larger area so that you can see the full content of your notes.

Before we go to the second Icon go to **General>Frequently Used Text** and create a common Text Message such as "**This Item comes with a 36 Month Warranty covering Parts and Labour**". Having done that go back to the Job Order Line and click on the second Icon. A separate screen will appear showing the Frequently Used Text message that you have just created. If you 'double click' on the selected text, to highlight it and click the **OK** button the text will be copied to these Notes.

8.4 Invoicing Options

Ostendo provides the following Invoicing Options:

- Invoicing Schedule linked to Planned Activities
- Invoicing Schedule linked to Fixed Price
- Invoice what was actually used
- No Invoice

The Invoice option is initially defined at the Job Type level. Go into **Jobs>Settings>Job Types** where you will see six Job Types already defined where each has an **'Invoice Style'** linked to it. We will refer to this during the following exercises.

1. Invoicing Schedule linked to Planned Activities

Go into **Jobs>Job Orders** and click the **'Add'** button. A panel will appear. Select Job Type **'Progress'** (This has an Invoice Style **'From Schedule'** linked to it) and select Customer **'Jim Gold & Co Ltd'**. Click on the **'Create Order'** button to generate the Job Order. You will see, to the bottom-right of the **'Detail'** screen a section headed **'Job Values'**. We will be concentrating in this area during our analysis of the Invoicing Options

Go into the **'Lines'** tab and add a single line with Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **10** Note: This action is simply to create a Job Order containing a reasonable invoicable value.

If you now go back to the Job **'Detail'** tab you will see (in the **Job Values** area) that the Ordered Nett Value is **\$750** (Calculated from **10Hrs** of **'GENERAL TIME'** at **\$75** per hour)

Within this **'Job Values'** area, click on the **'Job Invoices'** button. This will bring up a new screen for you to generate a **'Planned'** Invoice schedule for the **\$750**. (Note: you can also access this screen via **Jobs>Job Invoices**).

The screen shows the current **Total Job Value** and the **Unscheduled Job Value** (I.e. The amount that has yet to be included in a Planned Invoicing schedule).

1.1. Retentions

Before we create an Invoicing Schedule let's look at the **'Detail'** screen. You will see that you have the option to identify a **Retention %** such that each Invoice generated will have the identified percentage withheld until a later date. Let us now prepare for holding a Retention.

If you go into **Inventory>Descriptors** you will see an entry called **'Retention Claim'**. The main reason for using a Descriptor is that its Tax Group combined with the Customer's Tax Group identifies the Tax Code (and therefore the Tax Rate) that will be charged. Save the record and then return to the Job Invoice **'Detail'** screen for the above Job.

Enter (say) a **Retention%** of **10%**. Choose a Retention Date of (say) **3** months into the future, and select **Descriptor 'Retention Claim'** that you have just created. Save the record and click on the **'Invoices'** tab and we will now generate an Invoicing Schedule.

1.2. Invoicing Schedule

Go into **Inventory>Descriptors** and create a new entry **'Progress Claim'**. As above, the main reason for using a Descriptor is that its Tax Group combined with the Customer's Tax Group identifies the Tax Code (and therefore the Tax Rate) that will be charged. Save the record and then return to the Job Invoice **'Invoices'** screen for the above Job.

Click the '**Add**' button and a scheduled invoice will be generated in the lower part of the screen. Under field '**Code**' of the generated record select '**Progress Claim**' then '**Save**' the record. Let's now have a look at what has happened:

- A single Scheduled Invoice has been generated. I.e. It is still only '**Planned**' and has not yet been physically created.
- Its Value is \$**750** which represents **100%** of the total Job Value
- The **Unscheduled Job Value** field has been reduced to zero
- It has Descriptor '**Progress Claim**' which will print out on the actual Invoice ***

*** You should note that there 3 options available to define what is printed out on the Invoice. Click on the '**Invoice Line**' tab and let's look at these.

Use the Descriptor. This will print Descriptor '**Progress Claim**' along with its Description and Line Comments

Use the Descriptor with additional Breakdown. This will print Descriptor '**Progress Claim**' along with its Description and Line Comments. It will also print the contents that you have added via the '**Lines for Printing Only**' tab. The additional lines are for information only and will not affect the value of the Invoice. Add a line and select Descriptor '**Labour**' and complete the line as required. Repeat for Descriptor '**Material**' so that you now have two reference lines for printing

Use the Descriptor with additional Breakdown. This will not print Descriptor '**Progress Claim**' but will only print the contents that you have added via the '**Lines for Printing Only**' tab. Although the additional lines are for information only you should ensure that the total value of the lines equals the Invoice claim amount. Fields are available on the '**Lines for Printing Only**' screen to help you with this task.

Select the option that you are going to use for this scheduled Invoice.

Before you generate the physical Invoice go back to the '**Schedule**' tab and change the percentage in field '**Plan %**' to (say) **10%** then '**Save**' the record. You will notice that the '**Plan Value**' will reflect the reduced percentage and the residual amount will re-populate the '**Un-Scheduled Job Value**' field. Note: You may optionally amend the '**Plan Value**' rather than the '**Plan %**'

If you click the '**Add**' button a second Scheduled Invoice using the '**Un-Scheduled Job Value**' will be generated. Under field '**Code**' select '**Progress Claim**' then '**Save**'. You now have 2 '**Scheduled**' Invoices. However they both contain the same '**Plan Date**' so click on the drop-down calendar against the second Scheduled Invoice and amend the date to (say) **2** weeks from today. As previously stated you have only created an Invoicing Schedule for the Job, so now let's create an Invoice.

1.3. Generating the Invoice

There are two ways that Invoices can be generated:

- From within the above screen
- Batch Invoicing function.

We will use the first option to generate the Invoice. The second option will be addressed later in this exercise.

In the '**Schedule**' tab select the first Scheduled Invoice then click on the '**Create Pending Invoice**' button. This will generate an Invoice Number which will populate the '**Invoice No**' field. Note: The Invoice has still not been printed so you can delete the generated Invoice Number by clicking on the '**Delete the Scheduled Invoice**' button.

Now click on the '**Print Selected Invoice**' button situated to the bottom-left of the screen. By printing the Invoice you have now generated financial information and cannot now 'Delete' the Invoice. You can, however, click the '**Generate a Credit for the Selected Invoice**' button.

If you look at the generated Invoice you will see that:

- The Invoice lines reflect the print layout option
- If you have defined that Retentions are to apply to the Schedule then, at the bottom of the Invoice, you will see a Retention statement and amount.

2. Invoicing Schedule linked to Fixed Price

This again uses an Invoice Style of '**From Schedule**' but uses a manually entered value as the Price against which the Schedule is Generated.

'Go into **Jobs>Job Orders** and click the '**Add**' button. A panel will appear. Select Job Type '**Progress**' (This has an Invoice Style '**From Schedule**' linked to it) and select Customer '**Jim Gold & Co Ltd**'. Click on the '**Create Order**' button to generate the Job Order. You will see, to the bottom-right of the '**Detail**' screen a section headed '**Job Values**'. In that area you will see a Checkbox called '**This is a Fixed Price Job**'. If you 'check' that box then the adjacent field '**Nett Value**' will become active into which you can enter a value. Therefore enter a value of (say) **\$1000**. If you now click on the '**Job Invoices**' button you will see that field **Total Job Value** and the **Unscheduled Job Value** are populated with the Nett Value. The Invoicing Schedule as described above can now be generated using this Value as the basis.

3. Invoicing for Actual Time and Material

Go into **Jobs>Job Orders** and click the '**Add**' button. A panel will appear. Select Job Type '**Standard**' (This has an Invoice Style '**From Actual Entries**' linked to it) and select Customer '**Jim Gold & Co Ltd**'. Click on the '**Create Order**' button to generate the Job Order.

Go into the '**Lines**' tab and add a single line with Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **10** Note: This action is simply to create a Job Order containing a reasonable invoicable value.

If you now go back to the Job '**Detail**' tab you will see (in the **Job Values** area) that the Ordered Nett Value is **\$750** (Calculated from **10** hrs of '**GENERAL TIME**' at **\$75** per hour)

Within this '**Job Values**' area, click on the '**Job Invoices**' button. This will bring up a new screen that shows the current **Total Job Value** and the **Unscheduled Job Value**. These currently display **0** because we have not carried out any 'Actual' Bookings. Note: You can use the above 'Planned' value for Quoting and Estimating purposes but it has no effect on what you actually invoice.

3.1. Retentions

As with a '**From Schedule**' Style Invoice you can identify any Retentions that will apply to each Invoice. See 5.1.1.

Close the '**Job Invoices**' screen and go back to the Job Order that you created in 5.3.

3.2. Record 'Actual' Bookings

There are 3 methods by which you enter Job 'Actuals':

- Highlight the Job Line and Issue against the Line
- Click on an 'Issues' button on the Batch Entry bar of the Job Lines screen

- Go to **Jobs>Job Transactions** and issue the lines

Option 1. Highlight the Job Line and Issue against the Line

Go to the **'Lines'** tab of the Job Order and highlight **'GENERAL TIME'** in the upper part of the screen. In the lower part of the screen click on the **'Actual Issues'** tab. If you now click the **'Add'** button a new 'Actual' line will appear in the lower panel. Let's have a look at what this contains.

Qty. This is the Actual quantity that is issued to the Job. In our example enter a quantity of **6** and **'Save'** the record. If this line was an Item that had 'sub-level' variants (Colour, Size, Serial Number, etc) then you should click on the 'spyglass' icon and select the specific variant from the displayed list.

'Add' another line but this time enter a quantity of (say) **3** but don't save the entry until we have looked at other fields in the line.

Unit Cost. You have the option to amend the Cost of the booking if it is different to the current cost held against the record

Charge Style. By default the charge Style is **'Chargeable'**. However there are three variation of non-charge Styles. They can be selected from the drop-down list. These are:

- **Warranty;** If you select this Charge Style then you should also select the linked Warranty from the drop-down list under field **'Non-Charge Code'**. Warranties are taken from the list displayed under **Service>Warranty List**.
- **Contract:** If you select this Charge Style then you should also select the linked Contract from the drop-down list under field **'Non-Charge Code'**. Contracts are maintained under **Sales>Recurring Invoices**
- **Non-Charge:** If you select this Charge Style then you should also select the linked Cost Centre from the drop-down list under field **'Non-Charge Code'**. Cost Centres are maintained under **General>Cost Centres**

In this exercise select **'Non-Charge'** under **'Charge Style'** and (say) **'Warranty Costs'** under **'Non-Charge Code'**

Option 2. Click on the 'Issues' button on the Batch Entry bar

Go to the **'Lines'** tab of the Job Order and click on the **'Issues'** button on the **'Batch Entry'** bar. On the displayed panel you could prefill all 'planned' lines by simply clicking on the **'Prefill Issues'** button but what we are going to do is issue a Line to the Job that was NOT planned.

Click on the **'Add'** button and a new line will be presented in the main display. Make sure that the **'Line Type'** is **'Item Code'** and then select (say) **1500-2188** (Yellow Paint) with quantity **6** then **'Save'** the record. Click the **'Update Job with the Issues above'** button to the bottom of the screen. Upon confirming the bookings you will be taken back to the Job Lines screen where Item Code **1500-2188** will be displayed with a Planned **Order Qty** of 0 and an **Actual Qty** of **6**

Note: You can also click the **'Add'** button and issue a Descriptor or Item that was not planned for the Job Order. You will find that this will add a Job line but the **Order Qty** (The quantity that was planned) is **0** and the **Actual Qty** is the quantity that you have just issued. If the Job Invoice Style is **'From Actual Entries'** then this issue will be Invoiced

Option 3. Using Job Transactions

Go into **Jobs>Job Transactions** and create a new Batch. **'Save'** the Batch header and click on the **'Lines'** tab. Click on the **'Select Job for Prefilling Issues'** button and then select the

Job that you created above. It will display the 'Planned' quantity for the '**GENERAL TIME**'. You can alter this to reflect the 'Actual' time being booked. Note: You can also '**Add**' a new line an book 'Actuals' that were not Planned for the Job. Upon completion of the actual entries go to the '**Detail**' tab and click on the '**Post all Issue Entries**' button

If you go back to the Job '**Lines**' screen then you will see all the above 'Actual' values have been posted to the Job. Now click on the '**Detail**' tab so that we can generate the Invoice

In the Job '**Detail**' tab you will see (in the **Job Values** area) that the Ordered Nett Value is **\$750** (Calculated from **10Hrs** of '**GENERAL TIME**' at **\$75** per hour). You will also see that the 'To Be Invoiced' value comes from the Actual Bookings.

Within this '**Job Values**' area, click on the '**Job Invoices**' button. This will bring up a new screen for you to generate the Invoice. (Note: you can also access this screen via **Jobs>Job Invoices**). You will see that the current **Total Job Value** is the value of the Planned Lines and the '**To Be Invoiced**' amount is the current 'Actuals'. You could generate the Invoice in a similar manner to that carried out above but we will introduce a new function at this point

3.3. Invoice Approvals

Some businesses need to assess the Actual Bookings and 'Approve' then prior to generating the Invoice. To do this you should set up Ostendo as follows:

- Go to **Jobs>Settings>Job Rules** and 'check' the '**Invoice Approvals**' checkbox then '**Save**' the change. This means that ALL Planned Invoices (Scheduled or Actual Qty Style) must be approved before they can be generated into physical Invoices.
- If you go into **File>System Configuration>User Security & Options** and then highlight the User and click on the User Options tab you will see a checkbox called '**Allow Approvals**'. Make sure that this is 'checked'. This means that this User is allowed to Approve Invoices.

Now go to **Jobs>Job Pending Invoices** where you will see all 'planned' (Scheduled) and 'pending' (Actual Quantity) Invoices up to the current date are displayed. Amend the date to (say) 4 weeks in the future. You will see that the second Scheduled Invoice will now be presented. For each displayed Invoice click on the '**Details**' tab and 'Approve' the entry.

3.4. Generating the Invoice

As demonstrated earlier you can go into each Job and print the Invoice for the individual Job. What we are going to do now is generate a 'Consolidated' Invoice.

Go into **Sales>Customers** and select '**Jim Gold & Co Ltd**'. Go to the 'Detail' view and click on the '**Pricing and Invoicing**' tab (half way down the right-hand column). Select '**Consolidate by Customer**' from the drop-down list under field '**Invoice Consolidation**'. This means that all planned Invoices will be consolidated into a single Invoice.

Now go to Sales>Batch Invoicing and click the 'Add' button. On the presented panel:

- Amend the '**To Date**' to be (say) 4 weeks from today.
- 'Check' the '**Include Job Orders**' checkbox
- Select '**Jim Gold & Co Ltd**' from the drop-down against '**Specific Billing Customer**'
- Click on the '**Create**' button to create the Invoice

On the presented panel you may add any Freight charges and amend Credit Terms. However, you will also see, in the bottom panel, that two Jobs have been included in this single Invoice. Each Job is given a '**Source No**'. If you now click on the 'Lines' tab you will see what is being

Invoiced against which **'Source No'**. You have the option to 'Print' this Invoice from within the **'Detail'** view but we will use the Invoice Printing Function. Therefore **'Close'** the **'Batch Invoicing'** screen.

Go to Sales>Batch Invoice Printing and enter select **'Jim Gold & Co Ltd'** from the drop-down against **'Specific Customer'**, then click the **'OK'** button. On the panel that is presented you will see the Invoice that you have just generated. Select that Invoice and click the **'Print Current Invoice'** button. You have the option to bring the Invoice back to your screen. Note: If you select the **'Print All Selected Invoices'** button then the Invoice(s) will immediately print to your Printer.

If you view the Invoice you will see that it contains the multiple Jobs and each Job will have it's details printed.

3.5. Grouping and Summary Invoice Lines

There may be instances where you wish to group together similar lines such as Materials, Labour, or Miscellaneous activities. Also, within these groups, you may only wish to print a summary of the content (For example: All the various Employee Bookings can be displayed as a single charge line). To do this you should carry out the following:

Go into **General>Settings>Analysis Groups** and create a group called **'Labour'**. You will see three fields (Quote Form, Order Form, Invoice Form). Under each of these is a drop-down list that enables you to define the following options:

- **No Grouping:** All Lines in this Group will be printed
- **Header and Footer:** A Header containing the name of the Analysis Group will be printed followed by all Lines within the Group. At the end of the Group a Footer record will be printed showing a summarised total of the group content
- **Group Totals Only:** A single line will be printed showing the summarised total for the Group

For the purpose of this exercise select the following:

Quote Form: **Header and Footer**

Order Form: Not applicable to Job Orders (only Sales Orders)

Invoice Form: **Group Totals Only**

Now go into **Inventory>Descriptors** and select **'GENERAL TIME'**. In the **'Detail'** view select **'Labour'** from the drop-down list under field **'Analysis Group'**. Do the same against Descriptor **'MISCCHARGE'**

Go into **Jobs>Job Orders** and click the 'Add' button. Select Job Type **'Standard'** and select Customer **'Jim Gold & Co Ltd'**. Click on the **'Create Quote'** button (NOT the Create Order button) to generate the Job Quotation.

Go into the **'Lines'** tab and add:

A line with Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **50**

A line with Line Type: **Descriptor Code**, Code: **MISCCHARGE**, and Order Qty: **150**

If you go into the **'Detail'** view and click on the Quote 'Print' button you will see that each line is printed along with the 'Header' and 'Footer' of the Analysis Group. Whilst still in the **'Detail'** view click on the **'Convert Quote'** button and convert the quote to an Order.

Go into the **'Lines'** view and click on the **'Issues'** button in the **'Batch Entry'** bar then Issue both lines. Go back to the **'Detail'** view and click on the **'Job Invoices'** button then generate and print the Invoice. Note: You may find that the **'Create Pending Invoice'** button is 'greyed out' this is

because 'Approvals' is switched on. Go into **Jobs>Settings>Job Rules** and switch this off.

On the printed Invoice you will see that the line is simply a summary of the Analysis Group '**Labour**'

4. Retention Invoice generation

If you go back to the Retention screen against the Invoicing Schedule you created in 3.1. you will see that you can simply print the Retention Invoice from that screen.

Alternatively, as the Retention has a Planned Invoice date held against it, you can generate it using the Batch Invoicing routine described in 3.4.

Using either of these methods print the Retention Invoice.

8.5 Creating and Monitoring a Quotation

1. Quotation Styles

Within Ostendo you can use any one of 3 pre-defined Quote Styles.

- **Formal:** This shows all Line information along with each line's Unit Price, quantity and extended Price. All this is summated into a single Quote Price
- **Letter Style:** This is in the forma of a Letter that contains an introductory text (such as "Thank you for giving us the opportunity....etc) followed by a single line showing the Quoted Price and ending in a 'footer' Text (such as "This Quote is valid for a period of 30 days.....etc)
- **Letter Style plus Job Line information:** A combination of the above two Quote Styles

In preparation for generating the various styles go into **General>Frequently Used Text** and make two entries such as:

Name: Quote Header

Text/Phrase: Thank you for giving us the opportunity to make a Quotation. I'm sure that you will find our quoted price very competitive.

Name: Quote Footer

Text/Phrase: This Quotation is valid for a period of 30 days from the above date. If you have any questions please feel free to contact us at any time.

2. Quotation Settings

Ostendo has default settings for a Quotation that will be used to prefill the Quote Layout. You can, of course, amend make changes when printing the Quote. Go to **Jobs>Settings>Job Rules** where you will see the following fields that relate to a Quote

Quote Expiry Days: The number of days that the quote will remain valid. Currently this is set to **30** but you can amend this if required

Default Quote Header Notes: From the drop-down list select the '**Quote Header**' that you created above.

Default Quote Footer Notes: From the drop-down list select the '**Quote Footer**' that you created above.

Default Quote Style: Leave this as '**Formal**'

Default Quote Print Lines: From the drop-down list select the '**Print Single Line**' that you

created above.

Default Quote Line Description: Enter a short text such as **'We are pleased to offer a quote of:'**

'Save' the changes

3. Creating the Quotation

3.1. For 'Actual Qty' Invoice Style Jobs

Go into **Jobs>Job Orders** and click the **'Add'** button. Select Job Type **'Standard'** and Customer **'Jim Gold & Co Ltd'** then click the **'Create Quote'** button. Go into the **'Lines'** tab and add two lines:

Line Type: **Descriptor Code**, Code: **GENERAL TIME**, and Order Qty: **50**

Line Type: **Descriptor Code**, Code: **MISCCHARGE**, and Order Qty: **150**

then return to the **'Detail'** view.

You will see the following fields that are specific to a Quotation.

Quote Status: The Status is **'Quote'** and cannot be changed except by the program

Quotation Print Status: Field 'Quotation Print Status' has an adjacent button from which you can print the Quote

Quote Expiry: This is calculated from the System Date using the number of days set up in **Jobs>Settings>Job Rules**

Quote Opt: This allows you to adjust the content of the Quote.

Click on the **Quote Opt** button and a separate panel will be displayed. This shows the defaults that you set up above. Let us now look at the various Quote Styles that can be printed.

Formal: If you click on the **'Print Quote'** button then you will see the **Formal** style Quote. You should note that the lines are grouped by the Analysis Group you defined 5.3.5.

Go into **General>Settings>Analysis Groups** and, against **'Labour'** Group click on the drop-down against field **Quote Form** and select to **Group Totals Only**. Re-print the Quote and see the difference in the display.

You may wish to repeat the exercise and see the effect if the Analysis Group was set to **'No Grouping'**

Letter Style 1: Select **'Letter'** from the drop-down against **'Quote Style'** then **'Save'** the setting. If you now click on the **'Print Quote'** button then you will see the **Letter** style Quote.

Letter Style 2: You can see Style 2 by selecting **'Print Detail Lines'** from the drop-down under field **'Lines to Print'**. Now **'Save'** the setting and click on the **'Print Quote'** button. You will see that this is a combined **'Formal'** and **'Letter'** Quote

User: The 'User' style is available for your Administrator to create a radically different Style without affecting the 'Formal' or 'Letter' Quote Styles. Currently this contains a copy of the 'Formal' quote style.

3.2. For 'Schedule' Invoice Style Jobs

A Quote from a **'From Schedule'** Invoice Style Job Order has a slightly different Layout. In this instance it uses the Invoicing Schedule as the printed Lines on the Quote rather than the actual Job Lines. To demonstrate this go into the **'Detail'** screen and amend the Invoice Style to **'From Schedule'**. Because it is printed from an Invoicing Schedule you should first create the schedule as described in 5.1.2. then return to the **'Detail'** screen

Click on the **Quote Opt** button and let's have a look at the Quote Styles that can be printed for a 'From Schedule' Invoice Style Job.

Formal: If you click on the '**Print Quote**' button then you will see the **Formal** style Quote. In this instance the details of the Quote is taken from the Invoicing Schedule you have just created

Letter Style 1: Select '**Letter**' from the drop-down against '**Quote Style**' then '**Save**' the setting. If you now click on the '**Print Quote**' button then you will see the **Letter** style Quote.

Letter Style 2: You can see Style 2 by selecting '**Print Detail Lines**' from the drop-down under field '**Lines to Print**'. Now '**Save**' the setting and click on the '**Print Quote**' button. You will see that this is a combined '**Formal**' and '**Letter**' Quote

4. Monitoring the Quotation

Various Reports and Views are available to monitor Quotes. Take a look at the following:

Reports: Go to **Jobs>Job Reports** and have a look at the following Reports

Job Quote Listing: Enter '**Expiry Date To**' as yesterday to view expired Quotes

Job Quote Values Chart: Shows Converted, Current and Lost Quote values

Views: Go to **Jobs>Job Views** and have a look at the following.

Analysis - Job Orders: Select '**Order Status**' '**Lost**' and/or '**Quote**'

Customer Analysis: Select '**Order Status**' '**Lost**' and/or '**Quote**'

Go to **Help>Reference** and select **Global Options>Views Analysis Options** where you will find guidance as to how the Analysis View can be manipulated to give you the results you want.

5. Converting the Quotation

A Quote can either be converted into a Job Order or can be flagged as 'Lost'.

5.1. Convert to a Job Order

Go into the Quote you created above and, in the '**Detail**' view click on the '**Convert Quote**' button. On the presented panel 'check' the '**Converting to Order**' Radio button then click the '**OK**' button. The Quote will be immediately converted to a Job Order with status '**Open**'

5.2. Convert to a 'Lost' Status

When converting a Quote into a '**Lost Quote**' you are required to select a reason why the Quote was lost. These reasons are user maintained by going into **General>Settings>Quote Lost Reasons**. Go into that screen and add a couple of reasons to the current list.

Now create a Job Quote as described in 6.3. In the '**Detail**' view click on the '**Convert Quote**' button. On the presented panel 'check' the '**Quotation is Lost**' Radio button. The field underneath will become active where you can select the reason why it was lost from the drop-down list then click the '**OK**' button. The Quote will immediately be given a status of '**Lost**'.

8.6 Receiving an Item from a Job Order

This feature allows you to receive an Item from a Job Order and is primarily focused on the Repair and Refurbishment environment. The Item to be Repaired, Refurbished or Upgraded can (optionally) be issued to the Job and additional work recorded. The Item is then returned to Inventory along with the additional costs.

Some industries (Example: Boat refurbishment) maintain the Sell Price of the product as the Inventory Value. Therefore Ostendo provides a User-defined option to value this Item in Inventory at Cost or at its Sell Price

Let's see how this works.

1. Preparation

Job Type: Go into **Jobs>Settings>Job Types** and create a Job Type '**Refurb**'. It should have an '**Invoice Style**' of '**No Invoice**' and the '**Create an Item**' checkbox should be 'checked'. You should also note that you can define how the Item is valued (Cost or Price) under field '**How Item is Valued**'

Item: Go into **Inventory>Items** and create an Item '**Boat**'. Ensure that it is flagged as Serial Number Controlled. Next go into **Inventory>Inventory Adjustments** and create a new batch. Add this Item and receive (say) **1** into stock. (You will need to enter a Serial Number)

2. Adding the 'Received' Item to the Job

Go into **Jobs>Job Orders** and create a Job using the above Job Type '**Refurb**'. Go into the '**Lines**' tab and click on the '**Issues**' button in the Batch Entry Bar. You can now issue the '**Boat**' to this Job. You should also issue (say) **10** hours of Descriptor '**GENERALTIME**'.

Go back to the '**Detail**' panel and 'check' the '**Create an Item**' checkbox and then click on the adjacent '**Item Info**' Button. Enter Item '**Boat**' along with the Serial Number that you just issued to this Job.

You should note that the Item is automatically received back into Inventory at the time the Job Order is '**Closed**' therefore change the Job '**Status**' to '**Closed**' and '**Save**' the record.

If you now go into **Inventory>Inventory Availability** you will see that the **Boat** has been both issued to the Job and received from the Job. If you now go into **Pricing>Item Costing** you will see that the Cost of the Item has been adjusted to take account of the additional work

8.7 Jobs and Task Names

Let us commence this exercise by repeating what you did in Exercise 4. I.e. Go into **Jobs>Job Orders** and clicking the '**Add**' button. A panel will appear. Select Job Type '**Progress**'. Select Customer '**Jim Gold & Co Ltd**' then click the '**Create Order**' button. Add a couple of lines

If you look more closely at the Lines you will see that EVERY line in a Job Order is linked to a **Task**. A Task is simply a description of what actions are required using the Item(s), Descriptor(s), and/or Labour Code(s) linked to the Task. The Lines that you have just created you have been allocated to Task '**Job**'. Lets have a look at Tasks in more detail and how they are used

1. Creating Task Names

Go to **Jobs>Settings>Task Names** where you will see a pre-defined list of Tasks. You should also see that the Task '**Jobs**' has been defined as the 'Default' meaning that it will be used if no other Tasks are defined when creating a Job. Add a Task of your own (Example: **Commissioning**)

Go back to the above Job Order and, in the '**Detail**' view, click on the '**Job Tasks**' button. A panel will appear showing details about Tasks used in this Job. Click the '**Add**' button and add the Task that you have just created. '**Save**' the entry and exit the '**Job Tasks**' screen. (We will come back

to this screen in later exercises.). Go into the Job's **'Lines'** view and add Item **110-2039** (Washer-Stainless Steel-14mm). In the lower part of the screen click on the drop-down against field **'Task'** and select the second Task that has been included in this Job

Go back to the **'Detail'** view and click on the **'Print'** button against **'Job Sheet Status'**. On the generated Sheet you will see that the Order Lines are segregated into the Tasks where they are to be used.

8.8 Jobs and Task Bills

A Task Bill is a pre-defined list of materials, Descriptors and Labour activities. This is often referred to as a 'Schedule of Quantities'. It is used as the basis for adding a known Activity to a Job Line. The Task Bill has Header Information plus details lines that make up the Schedule of Quantities.

The following will go through creating Task Bill called **'Replace Wheel'** then creating a Job using this Task Bill

1. Creating a Task Bill

Go into **Inventory>Descriptors** and create two Descriptors:

Descriptor 1: Create a Descriptor called **'Callout Charge'** and complete the following fields:
Unit: '\$'
Description: **Standard Callout Charge**
Std Sell Price: **20**

Descriptor 2: Create a Descriptor called **'Replace Wheel'** and complete the following fields:
Unit: **'Each'**
Description: **Replace Wheel on Car**
Std Sell Price: **40**
'Check' **'The Descriptor is used as a Task Bill Code'** Radio Button

'Save' the records and exit the Descriptor screen

Go into **Jobs>Task Bill** and click on the **'Add'** button. In the displayed panel select **'Replace Wheel'** from the drop-down list. You will see that the **'Standard Sell Price'** will come from the Descriptor record. We now need to identify what Materials and Labour we will use within this Task Bill. Click on the **'Lines'** tab and enter the following:

Line Type: **Descriptor Code**, Code: **Callout Charge**, and Order Qty: **1**
 Line Type: **Item Code**, Code: **300-2167**, and Order Qty: **4**
 Line Type: **Labour Code**, Code: **LAB-SERVICE**, and Order Qty: **.25**

What this means is that the Sell Price (Invoice Price) of the Task Bill will come from the **'Replace Wheel'** Descriptor but the Costs (Cost of Goods) will come from the Actual Issues of the 3 lines within the Task Bill.

Note: In the **'Detail'** tab you have the option to calculate the Sell Price of the **'Replace Wheel'** Descriptor from the content in one of two ways:

- Use the Sell Prices of the Task Bill Content, or
- Use the Cost of the Task Bill Contents with an applied Markup

In either case the evaluated Sell Price will update the Base Sell Price in the **'Replace Wheel'** Descriptor record

2. Creating a Job Order using a Task Bill

Go into **Jobs>Job Orders** and click on the **'Add'** button. A panel will appear. Select Job Type **'Progress'** and select Customer **'Jim Gold & Co Ltd'**. **DO NOT** click the **'Create Order'** button yet because, in this instance, we are going to select a Task Bill before generating the order.

Click on the **'Task Information'** tab at the top of the Job Order Creation screen. You will see that it is prefilled with the 'Default' Task Name. As mentioned above this is merely a description of the Task's activities. If you look to the second line you will see a field **'How Job Lines are related to this Task'**. If you click the drop-down you will see two options:

Multiple Lines can be linked to this Task: This is the default and when used allows you to add any number of lines to this Task. This option is what you have been using up to now

A Single Task Bill is linked to this Task: If this option is selected then the **'Task Bill Code'** field underneath becomes active. You can link a Single Task Bill to a Task. From the drop-down list select Task Bill **'Replace Wheel'**. Now click the **'Create Order'** button

You will see in the generated Job Order that the Order Value will be the Sell Price held against Descriptor **'Replace Wheel'**. If you click on the **'Lines'** tab a single line whose **'Line Type'** is **'Task Bill Code'** is displayed. You will also see - in the lower part of the screen - the following information directly related to Task Bills

- A new Tab **'Task Bill Contents'** is shown. If you click on this then you can amend the content and make it specific to this Order.
- A new Tab **'Task Completions'** is shown. This allows you to define when the Task is complete and therefore (in an 'Actual Qty' Style Job) make it available for Invoicing.
- The **'Actual Issues'** tab allows you to enter details of what issues have been made against the Task Bill's Components.
- The **'Job Line'** tab allows you to amend the Sell Price (if your User record **File>System Configuration>User Security & Options** allows you to do this) by adjusting:
 - The Cost (calculated from the content or manually entered - subject the checkbox **'Unit Cost is calculated from Contents'** is 'checked' or not)
 - The Margin/Markup calculated from the entry in those two fields. This can be manually amended by 'checking' the **'Price is changed with Cost, Markup or Margin'** checkbox.
 - You can go directly into the **'Unit Price'** field and overwrite the Sell Price

In summary, the Task Bill Descriptor defines the **Sell Price** for the Job Line whereas the contents of the Task Bill, when issued, determine the **Cost** of that line

8.9 Jobs and Templates

A Template is a pre-defined Job Order collection of Tasks, Task Lines and Resources. A Template is - optionally - selected when creating a new Job; the copied contents can then be amended as required.

The following will take you through an existing Template and then creating a Job using the Template

1. Creating a Template

To create a Template you should go to **Jobs>Templates**. You will see that a Template **'ComputerNetworking'** currently exists. Select this and go to the **'Tasks'** tab. Here you can see

four Tasks that have already been entered. We will use this Template shortly but, for now, let us create a Template.

1.1. Let us first create a new Task Name

Go to **Jobs>Settings>Task Names** and add the following Task Name

Task Name: Ostendo Training
Description: Train User in Ostendo
Department: Select 'Administration' from drop-down

1.2. We will now create a new Task Bill

Go into **Inventory>Descriptors** and create a Descriptor called 'Training' and complete the following fields:

Unit: Each
Description: Carry out Training
Std Sell Price: 100
'Check' 'The Descriptor is used as a Task Bill Code' Radio Button

1.3. Go into **Jobs>Task Bill** and click on the 'Add' button. In the displayed panel select 'Training' from the drop-down list. Click on the 'Lines' tab and enter the following:

Line Type: Descriptor Code, Code: GENERAL TIME, and Order Qty: 2

1.4. Go to **Jobs>Settings>Templates** and add the following Template Name

Template Code: Detailed Training
Description: Train User in Ostendo

Go to the 'Tasks' tab and click the 'Add' button then add the Task Name 'Ostendo Training'. Amend the field 'How are Invoice Lines created' to refer to a single Task Bill and then select Training.

2. Creating a Job Order using a Template

Go into **Jobs>Job Orders** and clicking the 'Add' button. A panel will appear. Select Job Type 'Progress' and select Customer 'Jim Gold & Co Ltd'. You will see under the Customer a checkbox called 'Use a Template for this Job'. 'Check' this and select 'ComputerNetworking' from the drop-down in the adjacent field. A panel will be presented from which you can select the specific Tasks to be added to this Job. Select all the Tasks and click on the 'Create Order' button.

In the generated Order you will see that the selected Tasks from the Template have been copied along with all the Lines in each Task.

If you go into the 'Lines' tab you can see - on the 'Batch Entry' bar - that there is a button 'Templates'. If you click on this you can add more Templates to this Job Order to create a more complex Job. Click on the 'Template' button and select 'Detailed Training'. You will see that you can build up a Job by selecting Tasks for multiple Templates.

8.10 Costs and Values

Job Costs and Values provide a comprehensive view of the current and projected status of a Job. It is dynamically built from each Job line and displayed at Task, Job and Project levels.

1. Task Costs and Values

Go into a Job Order and click on the '**Job Task**' button. On the presented panel highlight a Task and then click on the '**Task Progress**' tab in the lower part of the screen. This shows the following information relating to this Task

Costs:

- Task Planned and Actual Costs with simple evaluation of percentage complete based on these figures plus remaining cost to complete the Task
- Calculated Percentage complete based on each line within the Task (a more accurate completion assessment) with Projected final Task Cost. Facility is provided to temporarily override the calculated % percentage complete to re-assess the final Task Cost.

Hours:

- Task Planned and Actual Hours with simple evaluation of percentage complete based on these figures plus remaining hours to complete the Task
- Calculated Percentage complete based on each line within the Task (a more accurate completion assessment) with Projected final Task Hours. Facility is provided to temporarily override the calculated % percentage complete to re-assess the final Task Hours

2. Job Costs and Values

We can take the previous Task Costs and Values and apply this at a Job Level. Go into a Job Order and click on the '**Related**' button down the right-hand side and select '**Job Values**'. The presented panel shows the following information relating to this Job

Invoices:

- Planned Invoices (Price, Costs, and Margins)
- Invoiced to Date (Price, Costs, and Margins)
- Percentage of Price - based on Actual to Planned Invoices
- Percentage of Costs - based on Actual to Planned Invoices

Values:

 Evaluated from each Task's planned and actual values

- Hours (Planned, Projected, % variance)
- Price (Planned, Projected, % variance)
- Costs (Planned, Projected, % variance)
- Margins - Planned and Projected

Costs and Progress:

- Job Planned and Actual Costs with evaluation of percentage complete based on these figures plus remaining Costs to complete the Job
- Calculated Percentage complete based on each Task within the Job (a more accurate completion assessment) with Projected final Job Cost. Facility is provided to temporarily override the calculated % percentage complete to re-assess the final (Projected) Job Costs

Time and Progress:

- Job Planned and Actual Hours with evaluation of percentage complete based on these figures plus remaining Time to complete the Job
- Calculated Percentage complete based on each Task within the Job (a more accurate completion assessment) with Projected final Job Hours. Facility is provided to temporarily override the calculated % percentage complete to re-assess the final (Projected) Job Hours.

WIP Value:

- Values are posted to WIP for the Job as, and when they are incurred. Costs are taken out of WIP when an Invoice is raised. The resultant WIP value, therefore represent the Costs for the Job that have not yet been Invoiced.

You can click on the '**Reports**' button down the right-hand side and select Report '**Job Costing Sheet**'. If you run this report you will see that it contains all the Job Cost and Value Information PLUS all the planned and actual activities that contributed to the Costs.

8.11 Projects, Costs and Values

We can take the previous Costs and Values and apply this at a Project Level. Go into a Job Order '**Detail**' tab and click on the drop-down against field '**Project**' and select the Project '**SAMPLEPROJECT**' then '**Save**' the record. Now go to **Jobs>Projects** and select the same Project. The following information about the Project is dynamically maintained from information held against each on the included Jobs.

Jobs:

- Jobs included in the Project

Project Progress:

- Estimated Completion values against Costs and Hours

Project Values:

- Income (Project Budget, Job Planned Totals, Job Actual Totals)
- Costs (Project Budget, Job Planned Totals, Job Actual Totals)
- Gross Profit (Project Budget, Job Planned Totals, Job Actual Totals)

This is compared against your entered Project 'Planned' Income/Costs and Start/End Date.

If you click on the 'Project Progress' tab in the lower part of the screen you will see that the Projects anticipated Costs and Hours are displayed based on the current progress.

8.12 Using the Job Calendar

By using the Calendar you can enter Jobs directly into a calendar based on the availability of your Field Service staff, etc

1. Defining the Calendar

Go into **General>Settings>Work Times** and you will see the current Calendar. The Job Scheduling function uses this calendar to determine each Task's Start and End Date/Time. Amend the calendar if you wish then '**Save**' the changes

2. Using the Calendar View

Go into **Jobs>Job Calendar** and you will see a base Calendar View. This view allows you to see what resources (Employees and/or Assets) are required to perform the required tasks. The main screen is split into three sections

- The main panel shows Resources, when and on what Job they are required.
- The upper-right panel shows a calendar for you to select the displayed time periods
- The lower- right panel shows all Jobs yet to be scheduled

2.1. Main Panel

The Main Panel shows all Job Tasks scheduled for each day displayed. The tasks on display can be viewed:

- For single or multiple Employees and/or Assets

- Grouped by Resource within Date, Date within Resource, or Jobs by Date
- In Daily, Workday, Weekly, Monthly, or user-selected time periods
- With a horizontal Timeline
- Filtered by Customer, Project, Job Type, Job Order, or Department

You can create new Job Orders from within the Calendar in one of two ways:

New Job From Start Time: Move the cursor into the calendar under a Resource and click on a start time. Now right click and select '**New Job From Start Time**' from the list. The Job creation screen will appear prefilled with the Timeslot's Start-date and assigned to this Resource. Simply select the Customer and nominate a duration in the lower-left of the panel. Click the '**Create Order**' button to generate the Job. The generated Job/Task will immediately appear in the Calendar View.

New Job From Selected Time: Move the cursor into the calendar under a Resource and drag down to encompass (say) two hours. Now right click and select '**New Job From Selected Time**' from the list. The Job creation screen will appear prefilled with the Timeslot's Start-date and assigned to this Resource. Now click the '**Create Order**' button to generate the Job. The generated Job/Task will immediately appear in the Calendar View.

The following options are available in this screen for you to try.

Time Scale: To amend the timeslots of the visible panel simply 'right click' on the scale down the left-hand side and select the displayed time intervals.

Task Information: Each Task (The 'activity' displayed in the Calendar) can show the following Information

Job Number - Job Number of the Task

Start and End Time - This is contained within the created Job's rectangle and is only displayed if the start time and/or end time does not coincide with the Time Scale defined above

Task Description - If you double-click on the Job then the Task Details will appear. In the lower part of the screen you will see a tab called '**Task Instructions**'. In this field enter (say) '**Check out broken washing machine**' and click the '**Save**' button. '**Close**' out of the screen to return to the Calendar where you will see the entered details.

Multiple Resources per Task: Once again double-click on the Job to make the Task Details appear then select the '**Task Resources**' tab. In this screen you can add a second Resource. Therefore add another Employee to this Task then click the '**Save**' button. '**Close**' out of the screen to return to the Calendar where you will see that the Task is now assigned to two employees and If you move one Resource's Start and End time for a Task then the other Resource's timeline will also move.

Task Status: The Task Status is displayed as a coloured 'band' surrounding the Task. To see the assigned 'Band' colours right click on a Task and select '**Task Status**' and select a Status.

Tracking Code: The Task itself can be colour-coded to provide an immediate view of the Tracking Status. You can define Tracking Codes by going into **Jobs>Settings>Tracking Codes** and creating a couple of codes along with their Colour Code. To assign a Tracking Code to a Task right-click on the Task and select '**Tracking Code**'. This will immediately change the colour-code of this Task to match the Tracking Code.

Drag and Drop a Task: You can drag a Task allocated to a single Resource and drop:

- Against another Resource
- In the same Resource but to another timeslot

Amend Timeline: You can amend the timeline against a Task dragging the Start or End boundary and dragging it to the Required Time

2.2. Calendar Panel

To the right of the screen you will see a panel that shows a Calendar by month. You can view more months by dragging the Calendar Panel boundary left/right/up/down to let you view the desired months. Within this panel you will see the following:

Active Days: Any Days that contain a Job Task will be in **Bold**. Clicking on this day will bring up that day in the Main Panel.

View Days: If you Drag the cursor over a few days then release, the Main Panel will immediately reflect the selected time-span.

2.3. Jobs Not Booked In Panel

At the lower right of the Calendar View you will see a panel that shows all the Jobs that have been generated (including the ones you created above) but have not yet been 'Booked In'. A Job must be 'Booked In' before it is scheduled and added to the Main Calendar Panel. To 'Book In' a Job you can either:

- Click on the '**Job Booking**' Button within a Job's Detail screen (**Jobs>Job Orders** 'Details' Tab)
- Highlight the Job in this panel then 'Right Click' and select '**Book In**'
- Double Click on the Job in this panel

Select one of the (**PRO*******) Jobs you created earlier and a '**Booking In**' panel will appear. In this panel carry out the following:

Enter the Date and Time that the job is scheduled to start
Click on the '**Calculate Job End Date and Time**' button
Click on the '**Book In**' button

The Job will now appear on the Job Calendar. However, you should note that we haven't allocated the Job to a Resource yet and, therefore the activity is (temporarily) held against all Resources. Simply right-mouse on any one of the generated activities and - in the drop-down list - select '**Assign to Resource**' and select the Resource that is going to do the Activity. Of course you can pre-assign a Resource when creating the Job and the Calendar will assign the Activity to the assigned Resource.

3. Job Task Scheduling

For the Job that you created from a Template in 9.2 (above) you will see that it has multiple Tasks. You can go to **Jobs>Job Orders>Job Tasks** then select the Job to view the Job Task details. (You can also click on the '**Job Tasks**' button in the Job Order '**Detail**' screen) The following details are taken from each Task when 'Booking In' the Job.

The Task '**Sequence**'
The '**Estimated Task Duration**'
The Task Resource(s) if defined under the '**Task Resources**' tab

To '**Book In**' a Job you can either go from the '**Job Calendar**' as described in 12.2.3. or you can click on the '**Job Booking**' button on the Job Order's '**Detail**' screen.

The '**Job Booking**' screen will be presented in which the Tasks are scheduled as follows

Using the **'Sequence'** held against each Task the lowest numbered **'Sequence'** is scheduled from the entered Start Date/Time and the Task duration applied using the Calendar (**General>Settings>Work Times**) and the end Date/Time of the Task is calculated. This becomes the start Date/Time of the next Sequence. If, however, the same Sequence Number is held against more than one Task then they are scheduled in parallel and the longer duration determines the start Date/Time of the next higher Sequence.

Having scheduled each Task they then require a Resource to be allocated to them. You can allocate the Resource prior to scheduling by going into the above **'Job Tasks'** screen and allocating the Resource(s) against each Task. Alternatively you can simply **'Book In'** the Job and allocate the Resource in the Job Calendar as described in 12.2.3

9 8. Service Orders

The Service function covers a wide-range of functionality from initial sale of a Serviced Item, through attaching a Preventive Maintenance schedule then carrying out Maintenance Events, to finally analysing Service History, Warranty Claims, etc

The following will be covered in this document:

- Warranty Definitions
- Service Plans
- Customer Asset creation
- Customer Service Scheduling
- Preventive Maintenance Jobs
- Breakdown Jobs
- Servicing History

9.1 Preparation

In preparation for using the Service function you should ensure that the following have been addressed.

1. Job Types

Whenever a Service Order is created it is given a Job Type. This allows Ostendo to focus on what the Order is intended to achieve and how it is to be invoiced. If you go into **Jobs>Settings>Job Types** you will see that a couple of Job Types relating to Service Orders have already been created. You should note that the specific options that relate to a Service Style Order are:

Job Style defines if the Order covers

Customer Asset (normally for Warranty and/or Servicing)

Company Asset (for Plant Maintenance or "In-house work")

Invoice Style relates to how this Job is to be Invoiced. The options are:

Scheduled: Based on agreed Price. Invoice(s) can optionally be pre-defined in an Invoicing Schedule

Actual: Based on Actual Usage incurred. This usage may be charged at Sell Price or Cost-Plus.

No Invoice: No Invoice required (Example - 'Company Plant' Maintenance)

Create your own Job Type if you want to use it in later exercises.

2. Templates

A Template is a pre-defined sequence of Tasks and Resources required to carry out each Task. If you go into **Jobs>Templates** you will see that a couple of Templates have already been created. For more info on Templates refer to the **Job Order Training guide**. A Template is linked to a Service Plan (see next paragraph) and is applied whenever the Service event is initiated. Create your own Template for use in subsequent exercises (or use the existing **AirConService**)

3. Service Plans

A Service Plan is a collection of regular Service Events that would apply to a Customer Asset. For example: the Asset may have a simple 3-monthly Service in addition to a more complex Annual Service. Go into **Service>Service Plans** where you will see a Service Plan (**STDAIRCONPLAN**) currently exists. Create your own Service Plan containing multiple future planned events; each containing:

Service Frequency: Days or Months

Length of Service Frequency: an entered number

Job Type: Use the above Job Type

Job Template: Use your generated Template or the current **STDAIRCONPLAN**

4. Warranty Definitions

Warranty Codes and their conditions can be maintained in Ostendo. The relevant Warranty Code(s) are applied to a Customer Asset. Each Warranty Code comprises of the following:

Warranty Code Identity: User defined

Term: Duration of Warranty

Coverage: Labour, Material, Labour and Material, or Special Details

Special Details: Identifying specific cover conditions

Exclusion: stating any exclusions

Extended Notes

Go into **Service>Warranty Codes** and view the Codes that have already been created. You may wish to create your own for use later in this Exercise

5. Asset Types

Asset Types are used to segregate Customer Assets for analysis purposes. Customer Asset Types are User-defined and can include such types as:

- Switchgear
- Vehicles
- Air Conditioning

If you go to **Service>Settings>Customer Asset Types** you will see that there are 3 Types currently in the system. Add another Type called '**Vehicles**'. We will use this later

9.2 Customer Asset creation

There are two types of Customer Assets:

- Pre-existing Asset against which you carry out Service Events. For example you may define a Customer's Building as the Asset against which you carry out regular service events such as Roof Cleaning, Checking Safety Equipment, etc.
- An Asset created when you sold the product. That Asset can have an after-sale Service Schedule linked to it. For example: Vehicle, Air Conditioning Unit, etc

1. Create Asset upon sale of an Item

Go to **Inventory>Items** and create a new Item (say) '**AC0001**' with Unit of '**Each**' and description of '**Air Conditioning Unit**'. On the Item's '**Detail**' screen carry out the following actions that are required for Items which convert into Customer Assets.

- 'Check' the '**Serial No**' checkbox to denote that it is Serial Controlled
- 'Check' the '**Create Customer Asset**' checkbox
- Click on the '**Customer Asset Options**' button and, on the displayed panel you should select the '**Asset Type**' that you created or viewed in 2.5. above. Also 'check' the '**Create Service Schedule**' checkbox and select Service Plan that you created or viewed in 2.3. above
- We will also identify this Item as being covered by a Warranty therefore 'check' the '**Sales Warranty Applies**' checkbox and select the specific Warranty Definition that you created or viewed in 2.4. above

The next step is to create Inventory records for this Item, therefore go to **Inventory>Inventory**

Adjustments and create a new batch. On the 'Lines' tab click the 'Add' button then:

- Click on the 'Item Code' field and select the Item you created above.
- Amend the 'Adjustment Type' to 'Receipt'
- Enter an 'Adjustment Qty' of 1
- Go to the 'Serial Number' field and enter a unique Serial Number for the Item

Repeat this and 'Add' another line for a different Serial Number of the same Item.

Finally go to the 'Detail' tab and click the 'Post All Adjustments' button to receive the two Serial Numbers into stock.

Now let's sell the Item.

Go to **Sales>Sales Orders** and select an existing Order that has a 'Type' of 'Counter Sales'. Go to the 'Lines' tab and click the 'Add' button. The cursor will be positioned in the 'Code' field where you should select the Item you created above then 'Save' the record.

You should now issue one of the serial numbers that you have created by clicking on the 'Picked Lines' tab in the lower part of the screen and clicking the 'Add' button to create a new pick record in the lower part of the screen.

Place the cursor in the 'Qty' field in this record then click the 'spyglass' icon that is within this field to display the current stock. Enter qty of 1 in the 'Issue Qty' field against one of the lines and then click the 'Save' button. Click the 'OK' button to exit the screen. The Item has now been issued so close out of the Sales Order

If you now go to **Service>Customer Asset** you will see that an Asset has been generated to cover this Issue. We will come back to this screen later.

2. Create Asset directly

Go to **Service>Customer Asset** and click the 'Add' button. Add a Customer Asset (say) vehicle registration **XYZ123** then click the 'Create' button

On the presented screen select Asset Type = 'Vehicles' and then select a Customer from the drop-down list under 'Customer' then 'Save' the record.

9.3 Other Asset information

1. Location at Customer Site

This allows you to denote where - at the Customer premises - the Asset is located. You should first define and set up the location as follows:

Create a Customer's Warehouse by going into **Inventory>Warehouses** and clicking the 'Add' button. Enter the following:

Warehouse Code: enter the new Warehouse Code

Description: enter a short description of the Warehouse

Company: Select 'Customer'

Name: Select the Customer who owns the Asset from the drop-down list

Next you should create the physical location within the above Warehouse by going to **Inventory>Locations** and clicking the 'Add' button. Add the Location to the above Warehouse.

Go back **Service>Customer Asset** and select the above Customer Asset. Click on the drop-down against field 'Location at Customer Site' button and select the Location that you have

just created.

2. Asset Tracking

This allows you to maintain a date and time-stamped history of where the Asset has been, and is currently, located

While you are still in **Service>Customer Asset** click on the '**Asset Tracking**' button. Click on the '**Add**' button and add a record defining where it is currently located.

3. Asset Additional Fields

There are two levels where Additional fields could be required against Assets:

- Global Fields that apply to ALL Assets (Example:- Service Level)
- Properties that apply to some Assets (Example:- Customer Ref No)

3.1. Global Additional Fields

Go into **File>System Configuration>Additional Fields** and click on the '**Add**' button. On the displayed line enter the following:

Module: Select '**Customer Assets**'

Caption: Enter the Additional Field name (Example '**Service Level**')

Field Type: From the drop-down list select the format of the field. Let us say that this is '**Text**'

Value List: This allows you to define any specific entries to which a drop-down list - during data entry - is restricted. In our exercise let's put two entries **Normal** and **Super**

'**Save**' the entry and '**Close**' the screen when done

If you now go to the Customer Asset screen (**Service>Customer Assets**) and click on the detail tab you will see a 'tab' (**Additional Fields**) to the left of the screen. Click on this tab and select the Service Level from the drop-down list then '**Save**' the record.

You can view the entries in the Customer Asset List screen if required by going into the List screen and 'right mouse' in the centre panel. Select '**Customize List Fields**' from the displayed panel. (Note: If that option is not visible then go to **File>System Configuration>User Security** and Options and go to the '**User Options**' tab for the current User. 'Check' both the '**Save Grid Layouts**' and '**List Customising**' checkboxes.)

On the displayed panel 'check' **Additional Field1** and give it a '**Display Name**' of '**Service Level**' and 'check' the '**Show Field**' checkbox. '**Save**' the entry. The field will now display on the List screen where you can sort and filter as necessary.

3.2. Customer Asset Properties

This feature allows you to define a 'Property' (Example: Customer Ref No) and then link that property to selected Customer Assets with a value that is specific to each Asset.

To demonstrate this go into **General>Settings>General Properties** and add '**Customer Ref No**' with Property Type of '**Text**'. Click on '**Save**' and then '**Close**'

Now go to **Service>Customer Assets** and select **XYZ123**. Click on the '**Related**' Button on the right of the screen and select '**Customer Asset Properties**'. Click the '**Add**' button and:

- Add a line using the above property
- Select '**Customer Ref No**' from the drop-down list under '**Property**'
- Enter the Customer's Reference Number that relates to this Asset

- 'Check' the '**Copy to Jobs**' checkbox

The selected property and value will accompany the Customer Asset whenever it is used in a Service Order. To demonstrate this, go into **Jobs>Job Orders** and click the '**Add**' button. On the presented panel select the following:

- Select '**Service**' under Job Type
- Select the Customer of the above Asset under '**Customer**'
- Select the above Asset from the drop-down under '**Customer Asset**'
- Click on the '**Create Order**' button to create the Service Order.

If you now click on the '**Related**' button down the right-hand side of the screen and select '**Customer Asset Properties**' you will see that the Asset Properties have been copied through to the Service Job.

4. Customer Asset Images

You can add multiple images (pictures, drawings, maps, plans, etc) to a Customer Asset. These can be printed on all documents where the Asset is used. Go to **Service>Customer Assets** and select the Asset you created above. Click on the '**Related**' Button on the right of the screen and select '**Customer Asset Images**'. Click the '**Add/Edit**' button and:

- Give the Image a short Name
- Point the program to where the image is located on your computer network

This image is now available for printing on any document referencing this Asset

5. Customer Asset Documents

You can add multiple documents to a Customer Asset. These can be printed along with all documents where the Item is used. Go to **Service>Customer Assets** and select the Asset you created above. Click on the '**Related**' Button on the right of the screen and select '**Customer Asset Documents**'. Click the '**Add/Edit**' button and:

- Give the Document a short Name
- Point the program to where the document is located on your computer network

This image is now available for printing on any document referencing this Asset

6. Customer Asset History Notes

This function allows you to link multiple time-stamped notes to a Customer Asset record. Against selected History Notes you can also add a dated reminder so that Ostendo will prompt you of the reminder once the date is reached. Go to **Service>Customer Assets** and select the Customer Asset that you created above. Click on the '**Related**' Button on the right of the screen and select '**Customer Asset History Notes**'. Click the '**Add**' button and:

- Enter some history notes
- 'check' the '**Follow-Up**' required checkbox and select a date from the adjacent drop-down calendar. '**Save**' the History Note and exit the screen
- To see the 'Follow Up' in action you should first change the company by clicking on **File>Change Company** and selecting another company then sign in as **ADMIN/pass**. Repeat this and go back to this company. Upon sign-in as **ADMIN/pass** the alert should present itself if the Follow-Up date is current

7. Customer Asset Hierarchy

The Customer Asset Hierarchy function allows you to link one or more 'Child' assets to a parent 'Asset' and view the Parent/Child relationships as required. To see how this is achieved go into **Service>Customer Assets>Customer Asset Hierarchy**.

1. Select a parent Asset and then click on the 'Detail' tab. Click the 'Add' button and select the Child Asset that is used on the Parent. Note: You can add many Child Assets to a single Parent.
2. For one the Child Assets that you have added go back to the 'List' view and select it. Now return to the 'Detail' screen and add Child Assets to this, thus creating a multi-level hierarchy of Customer Assets.
3. Finally, go back to the 'List' screen and select the Parent Asset used in 1 (above) and then click on the 'Hierarchy Tree' tab to see the multi-level Customer Assets.

9.4 Customer Asset Service Scheduling

Go to **Service>Customer Asset** and select the Customer Asset that was generated from sale of an Item (in Exercise 3.1). If you click on the 'Planning and History' tab you will see that the Service Schedule attached to the Item record has been converted into a Servicing Schedule against this Asset and each service event has a planned date taken from the delivery date of the Item using the periods defined in the Service Plan. We will be using this schedule in the following exercises.

1. Preparation

In the above Customer Asset click on the 'Detail' tab. On that screen you will see the following:

Checkbox '**Customer Requires confirmation of Planned Servicing**': The Customer may want reminders that the Service is scheduled so that they can confirm that the Service Date is acceptable. For our exercise you should 'check' this checkbox. This will open up the following fields, which support the Reminder function.

Days Notice Required: This is prefilled with the number of days prior to the Planned Service Date that the Customer requires to be notified. A system-wide default of **30** days is used but this can be amended and made specific to this Asset

Reminder Style: From the drop-down list select the style of reminder that will be used. The options are Phone, Email, or Letter. For our exercise select '**Email**'

Reminder Contact: From the drop-down list select the Contact at the Customers premises. If you cannot see any contact against this Customer then they can be created on one of two ways:

- Go to **Sales>Customers** and select the Customer then click on the 'Related' button down the right-hand side of the screen. In the drop-down list select '**Contacts**'. In the presented screen enter the Contact's details
- Go to **CRM>Contacts** and click the 'Add' button and enter the Contact's details

Confirmation Text: The 'Confirmation Text' converts user-defined text (including specific Customer Data) into a meaningful text when output to an email. You will see that one Confirmation Text currently exists.

We are confirming the planned service of your [MD_."CUSTASSETDESCRIPTION"] ([MD_."CUSTASSETNUMBER"]) due on the [MD_."PLANNEDDATE"]. [MD_."CONTACTFIRSTNAME"] this service is for [MD_."TEMPLATEDESCRIPTION"], please feel free to contact us if this time is not suitable.

Where the entries in [MD_.....] are fields that take data from the actual record being printed. The available fields are:

[MD_."CUSTASSETTYPE"] - Customer Asset Type
 [MD_."CUSTASSETNAME"] - Name of Customer's Asset

[MD_."CUSTASSETNUMBER"] - Asset Identity
 [MD_."CUSTASSETDESCRIPTION"] - Description of the Asset
 [MD_."CUSTOMER"] - Customer Name
 [MD_."PLANNEDDATE"] - Planned Service Date
 [MD_."JOBTYPE"] - Type of Job required to complete the Service
 [MD_."TEMPLATECODE"] Job Template that will be used to carry out the service
 [MD_."TEMPLATEDescription"] - Description of the Job Template
 [MD_."REMINDERCONTACTNAME"] - Contact Name held against the Asset record
 [MD_."CONTACTFIRSTNAME"] - First Name of Contact (from the Customer record)
 [MD_."CONTACTLASTNAME"] - Last Name of Contact (from the Customer record)

You may wish to amend the above text or create your own text by going into
[Service>Settings>Confirmation Text](#)

2. Service Confirmations

The next step is to review the Service Schedules against all Assets and, where the Customer requires confirmation, create the necessary reminder record.

Go into [Service>Service Confirmations](#) where Ostendo will show all Service Confirmations required up to the system date. Most probably the Asset that you created in 3.1. will not be in this list because the first Service Event is not due for another 3 months.

If you amend the date to include your Asset's first Service Event (I.e. Planned Service Date less the Reminder days) then you should see first scheduled Event. Select this event and click the '[Generate Reminders for selected confirmations](#)' button.

Whenever this button is selected then the following will happen for the various Reminder Styles

- **Email:** An email will be generated using the '[Confirmation Text](#)' identified against the Customer Asset.
- **Phone:** A report will be generated showing a list of all the Customers, Assets and contact details so that a telephone activity can be carried out. The Report List format is held under [File>Reporting Configuration>Report and View Developer](#) as '[Reminder List](#)'
- **Letter:** Each Customer will have a letter generated. The Letter format itself is held under [File>Reporting Configuration>Report and View Developer](#) as '[Reminder Letter](#)'

You can use the [Service>Service Confirmations](#) screen to monitor the status of Responses. To confirm a response and update the record select the Reminder record and click on the '[Detail](#)' tab. You update the Confirmation Status field and amend the confirmed Service Date. Therefore, for your Service Reminder that you have just sent go to the 'Detail' tab and amend the following fields:

Confirmation Status: [Confirmed](#)
Confirmed Service Date: Amend to [Today](#)'s date

3. Service Order Generation

The next step is to convert the Planned Service Events into Job Orders.

Go into [Service>Service Order Required](#) where Ostendo will show the following records whose scheduled 'Service Date' is equal to or earlier than the date selected in field '[Display Service Due By](#)':

- Service Events that do not require Customer Confirmation
- Service Events that have been confirmed
- Service Events that have not been confirmed

You can click on the checkbox at the top of the screen to exclude the third option

You should see the Service Event that you confirmed above. If you now **'Select'** this and click the **'Generate Job Orders for the selected Services'** button then a Job Order will be generated. Go into **Jobs>Job Orders** and view the generated **'Service'** order which uses the 'Template' that was attached to the Service Plan schedule

4. Customer Asset Record update

Go back to **Service>Customer Asset** and select the Asset against which the Job Order was created. If you go to the **'Planning and History'** tab you will see that the generated Job Order appears in the upper part of the screen

9.5 Breakdown / Repair Jobs

A Breakdown and/or Repair Job can be created as a standard Job Order using a Job Type whose Job Style is **'Customer Asset'**. Whenever this Job Type is used the program will ask for a Customer Asset to be entered. Having created the Breakdown or Repair Order you will find that the Customer Asset's **'Planning and History'** screen has been updated to record this Order.

9.6 Warranties

If you go to **Service>Warranty List** you will also see that a Warranty record has been generated to cover the Customer Asset you created in 3.1. using the Warranty Definition that you previously viewed or created 2.3.

This Warranty record links it directly to the Sales of the Asset in addition to defining the start and end date of the Warranty coverage

10 9. Suppliers and Purchase Orders

The Purchasing function covers the full spectrum from creating a Supplier through raising a Purchase Order and Receiving the Goods or Service, to Invoice receipt and matching.

10.1 Preparation

The following tables are used when creating Supplier records. Take a look at them. There are some defaults already set up but you may wish to add more or amend the current records:

Mandatory Tables

The following fields are mandatory and validated against a separate table when creating a Supplier record. Within each Table, however, you can nominate a 'default' that will populate a Supplier record when adding a new Supplier record.

Supplier Types: Segregates Suppliers into logical groups (e.g. Consumables, Hardware, etc). You can maintain these via [Purchasing>Settings>Supplier Types](#)

Tax Group: To facilitate Supplier/Item Tax code evaluation. You can maintain these via [File>Financial Configuration>Credit Terms](#)

Terms: Days from (Invoice, EOM, End of Next Month) plus Early Payment Discount. You can maintain these via [File>Financial Configuration>Tax Groups](#)

Optional Tables

The following fields are optional and, when used, are validated against a separate table when creating or maintaining a Supplier record.

Supplier Regions: For Purchase Analysis purposes. You can maintain Supplier Regions via [Purchasing>Settings>Supplier Regions](#)

Supplier Codes: A Supplier Code can be used to 'group' Suppliers within the same group (Example: Retail, Trade, etc). To maintain Supplier Codes go into [Purchasing>Settings>Supplier Codes](#)

Shipping Method: This allows you to pre-define that Shipping Method by which goods are received from the Supplier. To maintain Shipping Methods go into [Purchasing>Setting>Supplier Shipping Methods](#)

Buyer: You can allocate a default Buyer to a Supplier. This can be amended at Purchase Order level if required. To create a Buyer go into [Labour>Employees](#) and create an Employee. In the main Employee screen you can identify the Employee as being a Buyer

Currency Code: If you have Overseas Suppliers that use a different currency to your home currency than you should maintain the currency by going into [File>Financial Configuration>Currency Codes](#).

Note: There are a couple of ways you can create the Currency Symbol:

- Go into Microsoft Word and select Insert on the top toolbar. In the dropdown list select 'Symbol'. You can now select the symbol and 'Insert' into the word document. From there you can copy the symbol and paste it into this field
- Set your keyboard with 'Numerics Lock'. Using the enabled numbers due to the Numerics Lock being on (i.e. NOT the numbers 0 to 6 on the top row of the keyboard) you should place the cursor where the symbol is to appear and then - holding down the 'Alt' key - enter
0128 for the Euro (€)

0163 for the Pound (£)
 0165 for the Yen (¥)
 Add the Pound currency as we will use that later in these exercises

10.2 Create Supplier records

Go into **Purchasing>Suppliers** and add a new Supplier record. You will see that the created record is prefilled with the defaults identified in the previous section. These fields can be amended if required.

Having created the Supplier record the following linked records can be created and maintained

1. Supplier Additional Fields

There are two levels where Additional fields could be required against Suppliers:

- Global Fields that apply to ALL Suppliers (Example:- 'Preferred Supplier Flag', etc)
- Properties that apply to some Suppliers (Example:- Overseas 'Continent')

1.1. Global Additional Fields

Go into **File>System Configuration>Additional Fields** and click on the 'Add' button. On the displayed line enter the following:

Module: Select 'Suppliers'

Caption: Enter the Additional Field name (Example: Approved Supplier)

Field Type: From the drop-down list select the format of the field. The options are:

- **Text:** Any data format can be entered in a Text field
- **Decimal:** Allows entry of numbers and decimals
- **Integer:** Allows entry of whole numbers only
- **Currency:** Shows Currency symbol and decimals as defined in Regional Settings
- **Yes/No:** Shows a checkbox which can be checked/unchecked
- **Date:** Contains a drop-down calendar for selection of a date
- **Time:** Displays format HH:MM:SS for entry of a time of day

Value List: This allows you to define any specific entries to which a drop-down list - during data entry - is restricted

'Save' the entry and 'Close' the screen when done

If you now go to the Supplier screen (**Purchasing>Suppliers**) and click on the detail tab you will see a 'tab' (**Additional Fields**) in the centre-right of the screen. Click on this tab and 'check' this field then 'Save' the record.

You can view these additional fields in the Supplier 'List' view if required by going into the Supplier's 'List' screen and 'right mouse' in the centre panel. Select 'Customize List Fields' from the displayed panel. (Note: If that option is not visible then go to **File>System Configuration>User Security and Options** and go to the 'User Options' tab for the current User. 'Check' both the 'Save Grid Layouts' and 'List Customising' checkboxes.)

On the displayed panel 'check' **Additional Field1** and give it a 'Display Name' of (say) 'Approved Supplier'. 'Save' the entry. The field will now display on the List screen where you can sort and filter as necessary.

1.2. Supplier Properties

This feature allows you to define a 'Property' (Example: Service Level) and then link that property to selected Suppliers with a value that is specific to each Supplier (Example: Excellent, Good, Average, Poor)

To demonstrate this, go into **General>Settings>General Properties** and select **'Supplier'** from the drop-down under **'Module'**. Now add **'Service Level'** with Property Type of **'Text'** and the following entries - on separate lines - in the Property Values field (**Excellent, Good, Average, Poor**). Click on **'Save'** and then **'Close'**

Now go to **Purchasing>Suppliers** and select the Supplier you created above. Click on the **'Related'** Button on the right of the screen and select **'Supplier Properties'**. Click the **'Add'** button and:

- Add a line using property **'Service Level'**
- Select **'Good'** from the drop-down under column 'Value'
- 'Check' the **'Copy to PO Lines'** checkbox

The selected property and value will accompany the Supplier whenever a Purchase Order is raised against this Supplier. To demonstrate this, go into **Purchasing>Purchase Orders** and click the **'Add'** button. Select the Supplier then click the **'Create Order'** button. Click on the **'Related'** button in the generated Order and select **'Purchase Order Properties'**. You will see that the Supplier's Properties have been copied to the Purchase order.

2. Supplier Contacts

You can have multiple Contacts against each Supplier. To create these contacts click on the **'Related'** button when in the Supplier screen and select **'Contacts'**. (Alternatively you can go to **CRM>Contacts** and create the contacts via that screen using **'Contact Type'** = **Supplier** and selecting this Supplier under **'Company Name'**).

3. Delivery Addresses

Within Ostendo you can have an address for each Site within your company. One of those can be designated as the default address for Purchase Deliveries. You can create company 'sites' by going into **General>Company Sites**. Create your own site address of (say) **'Receiving Dock'**. If you now go into **File>System Configuration>System Settings** you can select **'Receiving Dock'** from the drop-down list under field **'Default Company Site'**. This will be used as the default delivery address for Purchase orders. You can, of course, amend this against specific Purchase Orders.

4. Supplier Images

You can add multiple images (pictures, drawings, maps, plans, etc) to a Supplier record. These can be printed on all documents where the Supplier is used. Go to **Purchasing>Suppliers** and select the Supplier you created above. Click on the **'Related'** Button on the right of the screen and select **'Supplier Images'**. Click the **'Add/Edit'** button and:

- Give the Image a short Name
- Point the program to where the image is located on your computer network
- 'Check' the **'Copy to Purchases'** checkbox then save and exit the screen

5. Supplier Documents

You can add multiple documents to a Supplier record. These can be printed along with all documents where the Supplier is used. Go to **Purchasing>Suppliers** and select the Supplier that you created above. Click on the **'Related'** Button on the right of the screen and select **'Supplier Documents'**. Click the **'Add/Edit'** button and:

- Give the Document a short Name
- Point the program to where the document is located on your computer network
- 'Check' the **'Copy to Purchases'** box then save and exit the screen

6. History Notes

This function allows you to link multiple time-stamped notes to a Supplier record. Against selected History Notes you can also add a dated reminder so that Ostendo will prompt you of the reminder once the date is reached. Go to **Purchasing>Suppliers** and select the Supplier that you created above. Click on the **'Related'** Button on the right of the screen and select **'Supplier History Notes'**. Click the **'Add'** button and:

- Enter some history notes
- 'check' the **'Follow-Up Required'** checkbox and select a date from the adjacent drop-down calendar. **'Save'** the History Note and exit the screen
- To see the 'Follow Up' in action you should first change the company by clicking on **File>Change Company** and selecting **DEMO** then sign in as **ADMIN/pass**. Repeat this and go back to company 'Training'. Upon sign-in as **ADMIN/pass** the alert should present itself if the Follow-Up date is current
- Note: In the CRM Module if a 'Call' was raised and subsequently 'closed' against a Supplier then any activity notes entered against that Call will be posted to the Supplier's History file

10.3 Supplier Catalogues

Multiple Supplier Catalogues can be imported and maintained within Ostendo. Normally Supplier Catalogue Items are **'Source on Demand'** such that whenever a demand originates from a Sales Order, Assembly Order or Job Order then a 'Demand' requirement is generated for you to immediately order the Item from the Supplier. You also have the option to hold Catalogue Items in stock. In this instance the demand will trigger a message asking if you would like to use the stock in preference to creating a Purchase order

1. Catalogue Importing

If you have a Supplier Catalogue then you can import it by going into Suppliers>Supplier Catalogues and clicking on the **'(Import/Update/Delete) Catalogue from File'** button and go through the import Wizard. Alternatively you may wish to use the Catalogue that is already loaded in the Training database.

The Import function goes through the following steps.

- Step 1:** point to the location of the Supplier's Catalogue
- Step 2:** Match Catalogue fields to Ostendo fields
- Step 3:** Define the format of the fields in the Import File.
- Step 4:** Import. Options are given to append, amend, add, etc
- Step 5:** Optionally save the above 'matching' for future use

2. Stocking Catalogue Items

You may wish to hold some Catalogue Items in Stock. Whenever a demand originates for the Catalogue Item and there is sufficient in stock to satisfy the demand then a message will be returned giving you the option to use the current stock. To demonstrate this go **Purchasing>Supplier Catalogues** and select an existing catalogue then click on the **'Lines'** tab. Against selected lines 'check' the **'Selected'** checkbox then click on the **'Convert selected codes to Inventory items'** button. If you now go to **Inventory>Items** you will see that the Item has been generated. If you click on the **'Detail'** tab you will also see that the:

- **'Default Supply Method'** is **'Source on Demand'**
- **'Sourced By'** is **'Purchasing'**
- **'Primary Supplier'** is the Catalogue Supplier

- 'Sell Price' and 'Buy Price' is taken from the Catalogue

Go into **Sales>Sales Orders** and select any Sales Order. Click on the 'Lines' tab and then click on the 'Add' button. In the lower part of the screen select 'Catalogue Code' under field 'Line Type' and select the above Catalogue Item. You will see a message come up if you wish to source from stock rather than the Supplier

3. Updating Prices on Stocked Catalogue Items

If the Supplier provides an updated Catalogue you can simply re-run the 'Import Routine' described in 3.1. to update the Catalogue records. However, in this instance you should use the option 'Append / Update: If record exists then update it otherwise add it'. You should also note that you need to identify the key fields for matching purposes. These are **CATALOGUENO** and **CATALOGUECODE**

You should also update the matching Supplier Items in Ostendo's Inventory. To do this 'check' the following fields in the 'Detail' view of the Supplier Catalogue Screen prior to carrying out the update

- Item Sell Price
- Item Buy Price
- Item Standard Cost

4. Catalogue Item Price Levels

A Sell Price matrix can be established against each Catalogue Item where a Sell Price and Quantity Discounts can be defined for user-defined Price Levels. A single Price Level is linked to a Customer record and whenever an Order is raised the Customer's Price Level price is used.

- You should first set up Price levels by going into **Pricing>Settings>Price Levels**. Try adding a couple of Price levels such as 'Trade' or "Wholesale"
- The next step is to create the Prices by Price Level. Go into **Purchasing>Supplier Catalogues** and click on the 'Price Level' tab in the 'Detail' screen. Add Sell prices for the Price Levels you have just created.

To see this in action

- Go into the Customer Screen (**Sales>Customers**) and select a Customer. Click on the 'Pricing and Invoicing' tab and attach one of the above Price Levels to the Customer
- Now create a Sales Order for the Customer by going into **Sales>Sales Orders**. 'Add' this Item to the Order Line. You should see the Price-Level price for the Item/Quantity come through to the Order Line

10.4 Supplier Prices

Against each Item or Descriptor you can maintain a Base Buy Price. Additionally you can maintain Buy Prices from multiple Suppliers. These take preference over the base price.

Each Supplier Price has the following information:

- Supplier
- Unit
- Supplier's Item Code
- Unit Price
- Up to 5 quantity-break prices

Option is given to restrict Purchase Orders to Suppliers in this Price List with optionally identifying a preferred Supplier.

1. Item Buy Price

Standard Buy Price

A Base Buy Price is held against each Inventory Item. To add the Base Buy Price go into **Inventory>Items** and select Item **100-2000** (Washer-Mild Steel-8mm). You will see that it already has a Buy Price in the '**Standard Buy Price**' field. Amend this if required. In the absence of any other Buy Price conditions (defined below) this price will be used. If no Price is entered then zero is assumed.

Supplier-based Buy Price

Against each Item facility is provided to add Supplier-specific Buy prices. This price is used when ordering the Item from that Supplier. To create the Supplier Specific 'Buy Price' go into **Inventory>Items** and select Item **100-2000** (Washer-Mild Steel-8mm). In the '**Detail**' screen click on the '**Pricing**' Button (Alternatively you can go into **Pricing>Item Pricing** to go to the same record). In the Pricing screen click on the '**Buy Prices**' tab. Click the '**Add**' button and enter a new line as follows:

Supplier: Select the Supplier to which this Buy Price will apply.

Unit: This is prefilled with the base Unit of measure for the Item but can be amended to reflect the specific Unit of Measure from this Supplier.

Supplier Item Code: You can (optionally) identify the Supplier's Item Number if it is different the above Item Number. Both numbers will appear on the Purchase Order.

Unit Price: Enter the Buy Price per Supplier's Unit of Measure.

Quantity Breaks: You can enter up to 5 Quantity Breaks

Now let's see how this works:

- Go into **Purchasing>Purchase Orders** and raise a Purchase Order against the above Supplier.
- Go into the '**Lines**' tab and add Item **100-2000**. You will see that the price is that defined against this specific Supplier.
- Go back to the '**Detail**' tab and click on the '**Print**' button. You will see both your Item Number and the Supplier's Item number in the Purchase Order.

One point to note is that you can enter a Supplier's Item Number when creating a Purchase Order Line. Let's see how this is done:

- Go to **File>System Configuration>System Settings** and make sure that the '**Advanced Searching**' checkbox is 'checked'
- Go back to the above Purchase Order '**Line**' screen and '**Add**' a new line. Click on the 'spyglass' icon against field '**Code**' and - in the displayed panel - enter the Supplier's Item Number (or even a partial number) in the 'Search' field. This should show you the Supplier's Item Number being linked to your own Item Number

Restrict to defined Suppliers

If you go into **Pricing>Item Pricing** for **100-2000** and click on the '**Buy Prices**' tab you will see a checkbox that can be set to restrict purchase from Suppliers in the Buy Price List. Any attempt to purchase this Item from other Suppliers will be rejected.

Primary Supplier

Against any Item you can identify a Primary Supplier. This is used during automatic ordering via the Replenishment run and prefills the Purchase Order's supplier with this Primary (or preferred) Supplier. Go into **Inventory>Items** and select **100-2000**. On the '**Detail**' screen select the primary (or preferred) Supplier

2. Descriptor Buy Price

Standard Buy Price

A Base Buy Price is held against individual Descriptors (Non-Inventory Items). To add the Base

Buy Price go into **Inventory>Descriptors** and select Descriptor '**GENERALTIME**'. Add a Buy Price in field '**Standard Buy Price**'. In the absence of any other Buy Price conditions (defined below) this price will be used. If no Price is entered then zero is assumed.

Supplier-based Buy Price

Against each Descriptor facility is provided to add Supplier-specific Buy prices. This price is used when ordering the Descriptor from that Supplier. To create the Supplier Specific 'Buy Price' go into **Inventory>Descriptors** and select Descriptor '**GENERALTIME**'. In the '**Detail**' screen click on the '**Pricing**' Button (Alternatively you can go into **Pricing>Descriptor Pricing** to go to the same record). In the Pricing screen click on the '**Buy Prices**' tab. Click the '**Add**' button and enter a new line as follows:

Supplier: Select the Supplier to which this Buy Price will apply.

Unit: This is prefilled with the base Unit of measure for the Descriptor but can be amended to reflect the specific Unit of Measure from this Supplier.

Supplier Code: You can (optionally) identify the Supplier's equivalent Code if it is different the above Descriptor Code. Both numbers will appear on the Purchase Order.

Unit Price: Enter the Buy Price per Supplier's Unit of Measure.

Quantity Breaks: You can enter up to 5 Quantity Breaks

Now let's see how this works:

- Go into **Purchasing>Purchase Orders** and raise a Purchase Order against the above Supplier.
- Go into the '**Lines**' tab and add Descriptor '**GENERALTIME**'. You will see that the price is that defined against this specific Supplier.
- Go back to the '**Detail**' tab and click on the '**Print**' button. You will see both your Descriptor Code and the Supplier's equivalent Code in the Purchase Order.

One point to note is that you can enter a Supplier's Equivalent Code when creating a Purchase Order Line. Let's see how this is done:

- Go to **File>System Configuration>System Settings** and make sure that the '**Advanced Searching**' checkbox is 'checked'
- Go back to the above Purchase Order '**Line**' screen and '**Add**' a new line. Click on the 'spyglass' icon against field '**Code**' and - in the displayed panel - enter the Supplier's Equivalent Code (or even a partial code) in the '**Search**' field. This should show you the Supplier's Code being linked to your own Descriptor Code.

Restrict to defined Suppliers

If you go into **Pricing>Descriptor Pricing** for '**GENERALTIME**' and click on the '**Buy Prices**' tab you will see a checkbox that can be set to restrict purchase from Suppliers in the Buy Price List. Any attempt to purchase this Descriptor from other Suppliers will be rejected.

Primary Supplier

Against any Descriptor you can identify a Primary Supplier. This is used during automatic ordering via the 'Create Required Orders' routine and prefills the Purchase Order's supplier with this Primary (or preferred) Supplier. Go into **Inventory>Descriptor** and select **GENERALTIME**. On the '**Detail**' screen select the primary (or preferred) Supplier.

3. Buy Price Updates

Two routines enable you to maintain Item and Descriptor Buy prices.

3.1. Batch Buy Price Update

This comprises of selecting a range of Items or Descriptors and applying a % change based on Standard Sell Price, Standard Buy Price, or Last Cost. To carry out the Buy Price Update go into **Pricing> Batch Price Update**. You can update either Items or Descriptors. Try updating the Buy Price against the Item or Descriptor you created above

3.2. Price Update from File

This function allows you to maintain standard Buy Prices or Sell Prices in some other database or spreadsheet and import these into Ostendo. This feature requires that the other database is capable of exporting to .csv or .xls format. The Price Update process uses a standard Ostendo routine to carry out this process and comprises the following Steps

- Create a Price Update Batch and define the update criteria
- Point the import function to the .csv or .xls file
- Match the import file to the fields in the .csv or .xls file
- Run the import function to a temporary file
- View the results and re-run if necessary
- Update the prices

To go through this process go to **Pricing>Price Update from File** and look at what is available. If you are going to go through this routine then please refer to the User Reference Guide.

10.5 Purchase Orders

1. Purchase Order Steps

A Purchase Order includes the following process flow:

- Order
- Receiving
- Invoice Receipt

The following flow options are available:

<u>Order Style</u>	<u>Order</u>	<u>Receipt</u>	<u>Invoice</u>
Full Process	x	x	x
Receipt & Invoice	-	x	x
Invoice Only	-	-	x

2. Purchase Order Source

Purchase Orders can originate from the following sources

2.1. On the Fly

An order can be raised for any Item or Descriptor by going into the Purchase Order screen and creating the Order. This is covered in more detail below. You can also create a Purchase Order for Items that are normally Assembled 'In-House'.

2.2. Source On Demand

Any Item or Descriptor whose supply method is '**Source on Demand**' will automatically have an Order Request created. A separate screen within the '**Replenishment**' function (See Inventory Control Training Guide) allows the user to view all Order Requests and optionally combine those requests into a single Purchase Order for the same Supplier.

2.3. Planned Order

A Planned Order is a Suggested Order generated via the Replenishment Run. A separate screen within the '**Replenishment**' function (See Inventory Control Training Guide) allows the user to view all Suggested Orders and optionally combine those requests into a single Purchase Order from a

Supplier.

3. Purchase Order Creation

3.1. Create the Order

Go into **Purchasing>Purchase Orders** and click the 'Add' button. A panel will appear. You will see that the Order can be created 'from scratch', or by copying an existing Purchase Order. Select Purchase Type '**Standard**'. And also select a Customer from the drop-down list then click the 'Create' button.

Look at the fields in the Order Header. You should note that the following are available against Purchase Orders and are similar in functionality to that which you addressed in Exercise 2 above

- Additional Fields
- Purchase Order Properties
- Purchase Order Images
- Purchase Order Documents
- Purchase History Notes

You should also note that you can amend the Delivery Address of the order to:

- Any Company Site Address, or
- Any Customer Address (Including Customer's Additional Addresses)

For further options refer to Ostendo Help

Click on the **Lines** tab to enter Purchase Order Lines

3.2. Add Purchase Order Lines

Many options are available for adding lines to the Purchase Order such as:

- Selecting from a user-defined List
- Selecting Items in batch
- Selecting Descriptors in batch
- Selecting multiple Lines from a Supplier Catalogue
- Selecting a single Item
- Selecting a single Descriptor
- Selecting a single Line from a Supplier Catalogue

3.2.1. *Selecting from a user-defined List*

Click on the '**List**' button on the '**Batch Entry**' bar. A panel is presented that displays all pre-defined Lists. You will see that a couple of Lists already exist in the database. You can create your own Lists via **Inventory>Lists**. Upon selecting a '**List**' a further panel will appear showing all lines in the List (Items, Descriptors, Labour). Select a couple of lines from the List and click the '**Create Lines from Selected Contents**' button. Each selected line will become a Purchase Order Line in its own right.

3.2.2. *Selecting Items in batch*

Click on the '**Items**' button on the '**Batch Entry**' bar. A panel is presented that displays all Items (excluding those with status 'Obsolete') in Ostendo. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the '**Add to Selected**' button

In the lower panel you can now amend the required quantity.

Once the full Item selection has been made then click the '**Create Lines from selected contents**' button to add the lines to the Purchase Order

3.2.3. Selecting Descriptors in batch

Click on the '**Descriptors**' button on the '**Batch Entry**' bar. A panel is presented that displays all 'Active' Descriptors that are designated 'for general purpose use'. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the '**Add to Selected**' button

In the lower panel you can now amend the required quantity.

Once the full Descriptor selection has been made then click the '**Create Lines from selected contents**' button to add the lines to the Purchase Order

3.2.4. Selecting multiple lines from a Supplier Catalogue

Click on the '**Catalogue Items**' button on the '**Batch Entry**' bar. If the Supplier has a Catalogue loaded into Ostendo then a panel is presented that shows all Items in the selected Catalogue. You can select lines, and place them in the lower panel, by either

- 'double clicking' on the line, or
- single click on the line and click the '**Add to Selected**' button

In the lower panel you can now amend the required quantity.

Once the full selection has been made from the Catalogue then click the '**Create Lines from selected contents**' button to add the lines to the Purchase Order

3.2.5. Selecting a single Item

This can be used as an alternative to the above 'Batch' selection. Click on the '**Add**' button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against '**Line Type**' select '**Item Code**' then go to the next field ('**Code**') to select the specific Item from the drop-down list. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select any Item.

3.2.6. Selecting a single Descriptor

This can be used as an alternative to the above 'Batch' selection. Click on the '**Add**' button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against '**Line Type**' select '**Descriptor Code**' then go to the next field ('**Code**') to select the specific Descriptor from the drop-down list. All the remaining information (except the 'greyed out' fields) can be amended as required. For the purpose of this exercise select any Descriptor

3.2.7. Selecting a single Line from a Supplier Catalogue

This can be used as an alternative to the above 'Batch' selection. Click on the '**Add**' button located to the right of the screen. This will allow you to add details of a single line entry to be made in the lower part of the Order Line screen. From the drop-down against '**Line Type**' select '**Catalogue Code**' then go to the next field ('**Code**') to select the Catalogue from which Items are to be extracted. If the Supplier has a Catalogue loaded into Ostendo then a panel is presented that shows all Items in the selected Catalogue. Select an Item from the Catalogue. Note: If the Supplier Catalogue Item also exists as an Inventory Item then a message will appear showing you the current stock of the Item and asks if you want to change the supply source to '

From Stock'

For the purpose of this exercise select Item **PT-3220** from the 'Power Tools' Catalogue

3.3. Additional Information in Purchase Order Line

Dependent upon the type of Purchase Order Line the following additional information is available

3.3.1. Line Info Band

Just above the lower detail panel you will see a band called 'Line Info'. The following buttons are presented in this band where applicable:

Alternate Items: If an Item has an Alternative Item referenced to it then this button is presented so that you can view information about that alternative. To see how this works let us suppose that Item **5000-2011** (Cat 6 Network Cable - 5 Metres) could be supplied as an alternative to Item **5000-2010** (Cat 6 Network Cable - 1.2 Metres). Go into **Inventory>Items** and select Item **5000-2010**. In the Detail view click on the '**Additional Inventory Settings**' button and 'check' the '**Alternate Item Available**' checkbox. Select Item **5000-2011** from the drop-down list in the field immediately underneath the checkbox.

Finally, using the Purchase Order created above, add '**5000-2010**' to the line. The Alternate Item button will now appear on the 'Line Info' band.

Stock: If any Order Line is an Ostendo Item then this button will be displayed. It shows the current On-Hand quantity (in stock), Available Quantity (On-Hand + Supply - Demand) plus the Item's basic Unit of Measure. Clicking on this button will show further details.

3.3.2. Additional 'Tabs' applicable to the Purchase Order Line

Line Allocations: This will show to where the line has been allocated. These can be the originating Order (Sales, Job or Assembly) or can be allocated to Inventory or destined to a Cost Centre. These allocations come from the source of the demand that created the Purchase Order Line. You should note that you can amend the Order Quantity and/or re-allocate that quantity by clicking on the '**Add**' Button. This will create a new line into which you can add the new Allocation destination. You will also see a button '**Edit Allocation Quantities**' just above the created line. Clicking on this button brings up a separate panel for re-allocation of Order Line's quantities. If the line is an Item Code and it has 'sub-level' variants such as Colour, Grade, Size then this feature allows you to optionally allocate specific variant(s) to where it is allocated.

Line Properties: Any specific property values held against an Item, Descriptor, or Catalogue Item are copied to the Purchase Order Line. You have the option to amend or delete current properties or even add new properties.

To see this in action you should first create the 'Property' via **General>Settings>General Properties** and add '**Voltage**' with Property Type of 'Text' and the following entries - on separate lines - in the Property Values field (**115 Volts**, and **230 Volts**). Click on '**Save**' and then '**Close**'

Now go to **Inventory>Items** and select Item **485-2267** (Internal Downlight 100 Watt). Click on the '**Related**' Button on the right of the screen and select '**Item Properties**'. Click the '**Add**' button and:

- Add a line using the above property
- Select the Voltage from the drop-down list
- Select the specific Voltage from the drop-down under column '**Value**'
- 'Check' the '**Copy to PO Lines**' checkbox

The selected property and value will accompany the Item whenever it is used in a Purchase Order.

To demonstrate this go into the Purchase Order that you were using above and select the **'Lines'** tab and add an Order Line for the above Item. Click on the **'Line Properties'** tab and the Properties will have been copied from the Item **485-2267**.

3.3. Order Line Notes

At the bottom of the 'Lines' screen there is space to put unlimited Order Instructions that apply to the specific Order Line. If you click in the Notes area you will see two Icons appearing in the top-right of the field.

If you click on the first Icon then the Notes field will occupy a much larger area so that you can see the full content of your notes.

Before we go to the second Icon go to **General>Frequently Used Text** and create a common Text Message such as **"This Item comes with a 36 Month Warranty covering Parts and Labour"**. Having done that go back to the Purchase Order Line and click on the second Icon. A separate screen will appear showing the Frequently Used Text message that you have just created. If you 'double click' on the selected text, to highlight it and click the **'OK'** button then the text will be copied to these Notes.

10.6 Purchase Order Receipts

Three options are available for Purchase Order Receipts

- Receive Goods or Services against a Purchase order
- Receive Goods or Services without a Purchase order
- Receive Goods that are included in a Shipment

1. Receive Goods or Services against a Purchase order

Go to **Purchasing>Purchase Receipts** and create a new receipt batch by clicking on the **'Add'** button. On the presented screen select **'Receipt for Order'** under **'Receipt Style'**. In the adjacent field select the Purchase Order that you created above then click on the **'Lines'** tab. All the un-receipted lines and their quantities will be displayed.

You now have two options:

- Click on the **'Prefill Receipt Quantities'** button to populate each line's **'Receipted Qty'** field with the quantity yet to be received against each line. You can amend the quantities as necessary.
- Go into each line and enter the quantity directly.

If you have a line that has pre-defined 'sub-level' variants (Colour, Grade, Size) along with the required quantity of each variant then the quantity entered above will be 'prorated' across the variants using the planned quantities. For example: If you planned to receive 10 of which 4 are to be Blue and 6 Red but actually received 5 then the quantities populating the variant receipts (click on the **'Allocations'** tab) will be 2 and 3 respectively. You should note that if you had received 6 then the 'Allocations' quantities will be 2.4 and 3.6. You should manually amend the quantities as required.

During this receipt process you also have the opportunity to adjust the receipt price.

Having completed the receipting process click on the **'Detail'** tab. If you have an Invoice that came with the receipt you can 'check' the **'Create Purchase Invoice for this Receipt on posting'** checkbox and an Invoice will automatically be generated. You will still have to 'post' the Invoice as described in 7.

Finally, click the **'Post all Purchase Receipt Entries'** button to 'Post' the Batch.

2. Receive Goods or Services without a Purchase order

Go to **Purchasing>Purchase Receipts** and create a new receipt batch by clicking on the 'Add' button. On the presented screen select 'Receipt No Order' under 'Receipt Style' then click on the 'Lines' tab. To receive a Line click on the 'Add' button and, in the lower part of the screen, enter an Item, Descriptor, or Catalogue Code that you are receiving. If you have purchased it specifically for an Order then click on the 'Allocations' tab and specify the destination of this receipt. You can also split the receipt across multiple destinations in a similar manner as you did in the previous receipt.

Once again, having completed the receipting process click on the 'Detail' tab. If you have an Invoice that came with the receipt you can 'check' the 'Create Purchase Invoice for this Receipt on posting' checkbox and an Invoice will automatically be generated. You will still have to 'post' the Invoice as described in 7.

Finally, click the 'Post all Purchase Receipt Entries' button to 'Post' the Batch.

3. Receipt Approvals

If you go into **Purchasing>Settings>Purchase Rules** you will see a checkbox called 'Purchase Receipt Approvals'. 'Check' this checkbox to denote that all Receipts must be approved before they can be accepted.

Obviously should carry out the approval process therefore go into **File>System Configuration>User Security and Options**. If you 'check' the 'Allow Approvals' checkbox then the User is allowed to carry out approvals via the 'Approvals' field described below. If this is not 'checked' then the User can see the 'Approvals' field but it is greyed out and access to it is not allowed.

Go back to **Purchasing>Purchase Receipts** and create a new receipt batch by clicking on the 'Add' button. On the presented screen you will see a field called 'Approval Status'. You can define the status of the receipt from the drop-down list. By default this is set to 'Waiting Approval'. It is only when this status is amended to 'Approved' does the 'Post All Purchase Receipt Entries' button become active and allow the next step to take place

10.7 Purchase Order Invoices

Two options are available for Purchase Invoices

- Create Invoice against a Purchase Order receipt
- Create Invoice without a prior Purchase Order receipt

1. Create Invoice against a Purchase Order receipt

Go to **Purchasing>Purchase Invoices** and create a new Invoice batch by clicking on the 'Add' button. On the presented screen select 'Invoice Matched to Receipt' under 'Invoice Style' and enter an Invoice Reference under field 'Supplier Invoice Number'. In the line below this select the Supplier against which the receipt was generated. Enter the Invoice Values and then click on the 'Lines' tab.

If you now click on the 'Select Receipt Batch for Matching' button a panel will be presented showing all the Receipt Batches that have not been matched. Select the Batch(es) that you wish to link to this Invoice then click the 'OK' button. All lines in the selected receipts will be displayed. Against your selected lines 'Check' the 'Matched' checkbox and enter the Invoice Qty. Note that you may also adjust the Invoice Unit Price if necessary.

Having entered the information relating to lines on the Invoice go back to the '**Detail**' tab where you should now ensure that the calculated values (in grey) from the Invoice Lines agree with the Invoice totals.

You will note in this screen that two fields (**Freight** and **Other**) are available for you to enter Freight and any additional charges that are on the Invoice. If you enter values there then you should note that:

- The costs entered are posted to Cost Centres identified in **Purchasing>Settings>Purchase Rules**
- These costs will not be included nor apportioned across the Invoice lines. If you wish to have this feature then you should use the '**Purchase Shipments**' function described below

Having completed the batch Invoicing process and balanced the actual Invoice Price to the Received lines you can 'check' the '**Post Purchase Invoice**' button to 'Post' the Batch.

2. Create Invoice without a prior Purchase Order receipt

Go to **Purchasing>Purchase Invoices** and create a new Invoice batch by clicking on the '**Add**' button. On the presented screen select '**Invoice Only**' under '**Invoice Style**' and enter an Invoice Reference under field '**Supplier Invoice Number**'. In the line below this select the Supplier from whom the Invoice was received. Enter the Invoice Values and then click on the '**Lines**' tab.

If you now click on the '**Add**' button you can add Item Codes, Descriptor Codes and Catalogue Codes and completing (at least) the following fields

- Line Type
- Code
- Invoice Quantity
- Unit
- Unit Price

Two points to note:

- If the Line Type is Item Code and the Item contains variants (Example: Colour, Size, Batch, etc) then you should click on the '**Allocations**' tab and specifically identify the Variants)
- If, in the Invoice Line, you entered an Allocation Type of '**Job Order**' or '**Assembly Order**' then the line and cost will be posted directly to the Order.

Having entered the information relating to lines on the Invoice go back to the '**Detail**' tab where you should now ensure that the calculated values (in grey) from the Invoice Lines agree with the Invoice totals.

Once again you will note in this screen that two fields (Freight and Other) are available for you to enter Freight and any additional charges that are on the Invoice.

Having completed the batch Invoicing process and balanced the actual Invoice Price to the Invoice lines you can 'check' the '**Post Purchase Invoice**' button to 'Post' the Batch.

3. Invoice Approvals

If you go into **Purchasing>Settings>Purchase Rules** you will see a checkbox called '**Purchase Invoice Approvals**'. 'Check' this checkbox to denote that all Supplier Invoices must be approved before they can be processed. This works in a similar manner to Purchase receipts Approvals. I.e.

Go into **File>System Configuration>User Security and Options**. If you 'check' the '**Allow Approvals**' checkbox then the User is allowed to carry out approvals via the 'Approvals' field described below. If this is not 'checked' then the User can see the 'Approvals' field but it is greyed out and access to it is not allowed.

Go back to **Purchasing>Purchase Invoices** and create a new Invoice batch by clicking on the '**Add**' button. On the presented screen you will see a field called '**Approval Status**'. You can define the status of the Invoice from the drop-down list. By default this is set to '**Waiting Approval**'. It is only when this status is amended to '**Approved**' does the '**Post All Purchase Invoice Entries**' button become active and allow the next step to take place

10.8 Overseas Suppliers

In this exercise we will:

- Create an Overseas Supplier
- Enter Buy Prices linked to that Supplier
- Raise a Purchase Order

1. Create an Overseas Supplier

Go into **Purchasing>Suppliers** and add a new Supplier record for (say) '**UK Exports**'. On the '**Detail**' panel:

- Amend the default Tax Group to '**NONTAXABLE**'
- 'check' the '**Foreign Currency**' checkbox and select **STL** under '**Currency Code**'.

All the remaining Supplier fields are as in Exercise 2.

2. Enter Buy Prices linked to the Supplier

Go into **Inventory>Items** and select Item **PT-3221** (Sanding Belt). You will note that it is a '**Purchased**' item. Change the '**Primary Supplier**' to the Supplier you have just created.

Click on the '**Pricing**' button to display the Pricing screen (you can also get there via **Pricing>Item Pricing**) and click on the '**Buy Prices**' tab in the middle of the screen. Click the '**Add**' button and add the above Supplier to the Buy Price list. You will notice the following:

- '**Currency Code**' is the currency of the Supplier
- '**Unit**' is the Unit of Measure that the Supplier will supply the Item
- '**Conversion**' converts the Supplier's UOM qty to the Item's base Unit of Measure
- '**Supplier Item Code**' is the normal Item Code used by the Supplier
- '**Supplier Description**' is the normal description of the Item used by the Supplier
- All the Buy Price fields use the **Currency Symbol** held against the Supplier

Add a Buy Price (which will be in the Supplier's Currency)

3. Raise a Purchase Order

Go into **Purchasing>Purchase Orders** and create a Purchase Order for Item **PT-3221** from the Supplier you created above. You will see that the Supplier Price comes through to the Purchase Order.

Add another Item to this Purchase Order. You will see that Ostendo uses the base Buy Price and has applied the currency conversion rate held against the Supplier's Currency.

Go to the **'Detail'** tab and print the Purchase Order back to your screen

10.9 Shipments

Shipments can cover shipments from local and overseas Suppliers. It provides facility to:

- Track the Shipment
- Add shipment costs
- Apportion the Costs across the Shipment
- Auto generate Invoice(s) to cover the shipment contents

In this exercise we will:

- Raise a Shipment and track its progress
- Add Shipment Costs and apportion them across the Shipment content
- Receive the Shipment's Contents
- Enter an Invoice to cover the Shipment

1. Preparation

The following tables are used to support Shipment records.

Shipment Type: This allows you to maintain Supplier Shipping Types, which are used to analyse Supplier Shipment records. Supplier Shipment Types cover both imports and inter-company transfers. Go into **Purchasing>Settings>Supplier Shipping Types** and create a **'Shipment Type'** of (say) **'Import'**

Shipping Method: This allows you to maintain Shipping Methods, which are used to analyse Supplier records. Go into **Purchasing>Settings>Supplier Shipping Methods** and create a **'Shipping Method'** of (say) **'UPS'**

Tracking Code: This allows you to maintain Shipment Tracking Codes, which enable you to track a Shipment through its delivery cycle. Go into **Purchasing>Settings>Purchase Shipment Tracking** and create a **'Tracking Code'** of (say) **'With Customs'**

Shipment Charge Code: This allows you to maintain the Types of Charges that could apply to a Shipment. These are maintained as Descriptors. Therefore go into **Inventory>Descriptors** and create a couple of Charge Codes such as (**Duty, Cartage, Storage**, etc).

2. Create Shipment Record and Track its progress

Go into **Purchasing>Purchase Shipments** and click the **'Add'** button to create a new shipment. Enter information into the **'Detail'** view plus 'check' the **'Shipment is for a single Supplier'** checkbox and select your Supplier in the field underneath the checkbox.

In the lower part of the screen select the **'Purchase Orders'** tab then click the **'Add'** button to add a new line then select the Purchase Order that you created above. Note: You have the option to amend the exchange rate if required.

You can track the shipment progress by manually updating the Shipment Details in the **'Detail'** view and adding shipment History notes by clicking on the **'Related'** button and selecting **'Purchase Shipment History Notes'**

3. Add Shipment Costs

In the lower part of the **'Detail'** panel select the **'Shipment Charges'** tab then click the **'Add'** button

to add a new line. Select an appropriate Descriptor (say '**Duty**') to represent the charge and identify the Allocation Method (the method by which charges will be apportioned to each line), the currency (leave blank if local currency) and the Duty charge for this shipment. Add another Line for (say) '**Storage**' and complete the fields as required.

You now have a shipment for a single Purchase Order and additional charges that apply to the shipment. What is now required is to apportion the charges across the Purchase Order Lines in the shipment.

4. Receive the Shipment's Contents and apportion Shipment Costs

Click on the '**Lines**' tab where you will see the value of each Shipment Charge and the apportioned amount to each Purchase Order line in the Shipment as calculated from the '**Allocation Method**' you defined for the Charge in the '**Shipment Charges**' screen.

You have the option to highlight any line and changed the receipt quantity, receipt cost and re-allocate the additional Shipment cost applicable to that line. Once you have completed any changes and all the '**Cost Differences**' are zero then you can click on the '**Detail**' tab and complete the Shipment. You now have two options:

If you have accompanying Invoices then you should 'check' the '**Create Purchase Invoices for all receipts on Posting**'. If you now click the '**Receive Purchases and Close Purchase Shipments**' button the receipt costs for each line will include the Shipment Charges and, if it is an Inventory Item then that will represent the Cost of the Receipt.

If you do not have accompanying Invoices then click the '**Receive Purchases and Close Purchase Shipments**' button the receipt costs for each line will include the Shipment Charges and, if it is an Inventory Item then that will represent the Cost of the Receipt.

In both the above instances Ostendo will create Purchase Order Receipt transactions.

- In the first instance they will automatically be 'matched' against the Invoice and 'Posted'.
- In the second instance it will have a status of '**In Progress**' and will require 'Posting'

11 10. CRM

The Customer Relations Management function allows you to take customer calls and either respond to them immediately or promulgate further action.

11.1 Preparation

The following tables are used within the CRM Function against Calls. There are some already set up but you may wish to add more or amend the current records:

Call Classification: Segregates Calls into logical groups for analysis purposes (e.g. Information, Quote, etc). You can maintain these via [CRM>Settings>Call Classifications](#)

Call Sub-Classification: Segregates Calls into logical sub groups within the Call Classification and is used for analysis purposes (e.g. Follow-Up, Confirmation, etc). You can maintain these via [CRM>Settings>Call Sub-Classifications](#)

Call Resolution Codes: Identifies the resolution identity of the call and is used for analysis purposes (e.g. Brochure Sent, Quote Given, etc). You can maintain these via [CRM>Settings>Call Resolution Codes](#)

11.2 Contacts

A table of 'Contacts' is maintained in Ostendo that covers Customer and Supplier Contacts (which can also be entered via the Sales and Purchasing) plus Prospects and Others that can only be maintained here.

1. Creating a Contact

To create a Contact go to [CRM>Contacts](#) where you will see that some Contacts already exist. Add you own contact and link that contact to (say) Customer '**Jim Gold & Co Ltd**'.

You should note that if the Contact has an email address then you can 'double click' on the address and it will bring up your PCs email facility prefilled with this email address.

2. Copy Contacts to 'Outlook'

You can copy the generated Contact's details to your Microsoft Outlook's Address Book by clicking on the 'Outlook' button to the right of the Contacts screen.

3. Contact Properties

This feature allows you to define a 'Property' (Example: Department) and then link this property to selected Contacts with a value that is specific to each Contact (Example: Buying, Accounts, etc)

To demonstrate how this works, go into [General>Settings>General Properties](#) and add 'Department' with Property Type of 'Text'. Click on 'Save' and then 'Close'

Now go to [CRM>Contacts](#) and select the Contact you created above. Click on the 'Related' Button on the right of the screen and select 'Contact Properties'. Click the 'Add' button and:

- Add a line using property 'Department'
 - Enter 'Accounts' in the 'Value' field
-

11.3 Call Centre

The Call Centre function enables you to:

- Register calls
- Locate an answer where possible
- Answer the Call, or
- Action a response from other personnel
- Monitor the Call
- Close the Call and maintain History

1. Registering a Call

Go to **CRM>Call Centre** and click on the 'Add' button to register a new call. Select the Contact that you created above. To help you progress the Call the following are available

If you click on the drop-down against 'Relates To' you will see that you can select various options. You should note that any selection except 'Call Only' allows you to link the call to a specific reference when clicking on the 'Linked To' field

If you select 'Job Order' or 'Sales Order' then the adjacent 'Create Order' button will become active. This feature allows you to go straight to those areas to immediately generate a Quote or Order from this Call.

Enter details of the Call in the 'Call Notes' field at the bottom of the screen

2. Finding an answer

In progressing the Call the following support facilities are available

2.1. Customer Statistics

Click on the 'Related' button to the right of the screen and select 'Customer Statistics'. This will provide current and past information about an existing Customer.

2.2. Knowledge Base

The Knowledge Base is simply a compilation of answers to questions that have been raised in the past and provide a library of knowledge that can be used in the Call Centre.

The following tables are used within the Knowledge Base function. There are some already set up but you may wish to add more or amend the current records:

Article Types: Article Types are used to analyse Knowledge Base records. These are User-defined and can include such types as:

- Education
- Setup
- How To Examples

Create your own Article Types and add them via **General>Settings>Article Types**

Article Categories: Article Categories are used when analysing Knowledge Base records. Article Categories are User-defined and can include such categories as:

- Internal Information
- Customer Questions

Create your own Article Categories and add them via **General>Settings>Article Categories**

Create a Knowledge Base record by going into **General>Knowledge Base** and clicking the **'Add'** Button. Enter (say) the following information into your first record

Question: How do we tell Ostendo to use the local email client rather than Microsoft Outlook?

Answer: Open Internet Explorer, click the Tools menu, select Internet Options. Click Programs tab and select your email client from the drop-down list. Once selected, click OK/Apply button.

Now go back to the Call that you created above. If you click on the 'Related' button to the right of the screen you can select **'Knowledge Base'**. Search facilities are available then enable you to quickly find any answers if they exist in the Knowledge Base

2.3. Price Inquiry

Quite often a Prospect or Customer may wish to know the current sell price of an Item or Descriptor. If you click on the 'Related' button to the right of the screen you can select **'Price Inquiry'** and enter the following

- Customer Name (I.e. Price Level known) or Price level
- Item or Descriptor identity (select from drop-down list)
- Quantity required (to take advantage of any Quantity Price Breaks)
- Unit of Measure (Base unit defaulted - amend from drop-down) for UOM discounts

Upon entering this information a display panel shows the Sell Price and all the applied discounts that enabled the price to be calculated

2.4. Inventory Inquiry

Quite often a Prospect or Customer may wish to know if you have stock of a certain Item and, if not, when you are expecting to receive it. If you click on the 'Related' button to the right of the screen you can select **'Inventory Availability'**. This will show, for a selected Item:

- Quantity held by Location
- Click on 'Transaction History' tab to view stock movements
- Click on 'Projected Availability' tab to view projected stock balances taking into account all supply and demand orders in Ostendo
- Click on 'Order Details' tab to view the current Supply and Demand orders

3. Action a response from other personnel

If you cannot answer the call yourself you can request further action by clicking on the **'Requires Further Action'** checkbox. This will activate the two fields to the right of the checkbox for you to identify the required response date and the priority. You will also find that a new tab **'Action Information'** appears just above the checkbox. Click on this tab to define the further actions.

Click on the **'Assigned To'** drop-down and select the Employee to whom this Call is being assigned. You can also select a cc Employee in the adjacent field. You can maintain employees via **Labour>Employees** and you should ensure that the selected Employee has an email address if you are going to email this Call Action Request

If you now click the **Email Action** button your PC's email facility will be activated and prefilled with Information about this call including the Call Notes you entered in 3.1.

4. Monitoring the Call Actions

If you go into **CRM>Views>Analysis - Call Tickets** you can select and filter the calls by status,

Priority, and required response date from the actioned Employee. Once a response has been received the response notes should be entered into the 'Action History' field in the Action Information tab

If the Action has been completed then 'check' the 'Action Completed' checkbox and enter the 'Sign-off date' and select the Employee who approved the activity completion

5. Reply to the Call

If the call has been resolved either directly or as a result of further actions, then you should identify the final call resolution details in the 'Call Resolution Information' tab along with the 'Resolution Code'

6. Closing the Call

If the Call is to be 'Closed' then simply click on the 'Close Call' button at the bottom of the screen and the Status of the Ticket will be changed to 'Closed'.

11.4 Service Level Agreements

A Service Agreement comprises of:

Steps that should be carried out in response to the Call

Severity (or Urgency) Levels that can apply to the Service Agreement

Response time required for this **Step/Severity**

	Urgent	Priority	Normal
Respond	1 minute	30 mins	1 hour
On Site	1 hour	2 hours	3 hours
Completed	2 hours	4 hours	6 hours

Each Customer can be allocated a Service Agreement and is given a Service Agreement Code and a Severity Level in the Customer record.

For monitoring your own performance you can identify a target achievement rate expected from each **Step/Severity**

	Urgent	Priority	Normal
Respond	100%	90%	90%
On Site	100%	90%	85%
Completed	100%	80%	70%

Let us see how this works

1. Setting up Service Agreements

1.1. Service Agreements

Go into **CRM>Settings>Service Agreements** and click the 'Add' button. Now create a new Agreement record called (say) 'Retail'

1.2. Agreement Steps

Go into **CRM>Settings>Service Agreement Steps**. Steps are used to provide 'milestones' against which the Call can be progressed and monitored. Add (say) three steps called **Respond**, **On Site**, and **Completed**.

1.3. Agreement Levels

Go into **CRM>Settings>Service Agreement Levels**. These are used to identify different Service (or Priority) Levels within the same Service Agreement. Add (say) three Levels called **Urgent**, **Priority**, and **Normal**.

1.4. Agreement Measures

We will now pull all the above together. Go into **CRM>Settings>Service Agreement Measures**. Against each combination of Agreement Step and Agreement Level within the Service Agreement you can identify

- The expected Response Time from the time the Call was first received
- The expected level of achievement between the Expected Response Time and the Actual Response time

	Urgent		Priority		Normal	
<i>Respond</i>	1 minute	100%	30 mins	90%	1 hour	90%
<i>On Site</i>	1 hour	100%	2 hours	90%	3 hours	85%
<i>Completed</i>	2 hours	100%	4 hours	80%	6 hours	70%

2. Service Agreement Activation

Now lets 'activate' Service Level Agreements. Go into **CRM>Settings>CRM Rules** and enter the following:

Service Level Tracking Activated: 'Check' the checkbox.

Default Service Agreement: From the drop-down list select the Default Service Agreement that will be used when generating a Call. As you have only created one Service level Agreement you should (of course) select that one

Default Severity Level: From the drop-down list select the Service Level within the above Agreement that will be used as the default when generating a Call. You should note that if a Customer has a Service Support Agreement/Service Level identified against the Customer record then that Service Agreement Level will be used in preference to this default. For now select Agreement Level '**Normal**'

3. Adding a Service Agreement to a Customer record

3.1. Adding the Service Agreement

Go into **Sales>Customers** and select a Customer (say) '**Jim Gold & Co Ltd**' then go to the '**Detail**' tab. Click on the 'Defaults' tab and enter the following:

Service Agreement: Select Service Agreement '**Retail**' from the drop-down list.

Service Agreement Level: Select '**Normal**' from the drop-down list.

3.2. Adding a Customer Contact

Whilst still in the Customer screen click on the 'Related' button and select 'Contacts' from the displayed list. (You can also go to this screen via **CRM>Contacts**). Add your own contact and link that to Customer 'Jim Gold & Co Ltd'. You should also note that if the Contact has an email address then you can 'double click' on the address and it will bring up your PC's email facility prefilled with this email address.

4. Using Service Level Agreements

Let us create a Service Call received from the above Customer. Go to **CRM>Call Centre** and click on the 'Add' button to register a new call. Select the Contact that you created above.

You will see two tabs that are visible when you enabled Service Level Tracking in the CRM Rules.

4.1. Service Level - Sub Tab

This shows the **Service Agreement** and **Service Level** held against the Customer record. If the Customer does not have a definitive Service Agreement or the source of this call is from other than a Customer then the default that you identified in CRM Rules will be used. Within this tab you can amend the following:

Severity Level: This is prefilled with the Default Service Agreement Level as entered into the CRM Rules screen or the Service Agreement Level as held against a Customer Record. It can be amended here to another Severity Level if required

Notes: You can enter unlimited amount of Notes that are related to this Service Level. If you click in the Notes area you will see two Icons appearing in the top-right of the field. If you click on the first Icon then the Notes field will occupy a much larger area so that you can see the full content of your notes.

Before we go to the second Icon go to **General>Frequently Used Text** and create a common Text Message such as '**This Severity covers Normal Service Agreements**'. Having done that go back to the Service Level - Sub Tab and click on the second Icon. A separate screen will appear showing the Frequently Used Text message that you have just created. If you 'double click' on the selected text, to highlight it and click the **OK** button then the text will be copied to these Notes.

4.2. Service Response - Sub Tab

This shows all the Steps contained within the Service Level Agreement for the chosen Service Level. This screen shows the Planned Response Date/Time based on the Call's Date and Time and scheduled against each Step based on the Work Times set up in **General>Settings>Work Times**. In this screen you can record the actual time spent on each Step thereby enabling you to analyse the actual response times relative to the agreed Service Agreement response times.

Go into field '**Actual Response Time**' and enter the actual Date/Time when the Step was completed. You can click on the drop-down Calendar and

- Select the Date by clicking on the Day
- Select the Time by clicking on the time element (HH, MM, or SS) and using the up/down arrows
- Selecting the a.m./p.m. field and alternating using the up/down arrows
- Click on the '**Now**' button to select the current Date/Time

Complete each Step with a genuine time so that we can review it in the next section.

11.5 Activity Events

1. Activity Events

Activity Events allow you to set up and maintain Activities and Employees and Company Assets required to support those activities. You can log Events as they arise and then:

- Allocate the Event to an internal Resource
- Add an Event Type and Status
- Determine an estimated start date/time and duration
- Define a Contact Name and details
- Add Activity Notes

Go into **CRM>Activity Events** and click on the 'Add' button. Add an Activity Event called (say) '**Property Valuation**'. In the remaining fields enter (for example) the following

Event Type: Select '**Business**'

Event Status: Select '**Tentative**'

Description: Enter (say) '**Valuation of Property and Outhouses**'

Resources: Select '**Bob Drum**'

Start Date / Time: Enter the start date and time for this Event.

End Date / Time: Enter the end date and time for this Event.

All Day Event: Leave unchecked

Where: A free-format entry field in which you can enter the Property address.

Contact Type: Select '**Customer**'

Contact Name: Select the Contact that you created in 2.1.

Message: At the bottom of the Activity Events screen there is space to put unlimited Notes that apply to the Event. If you click in the Message area you will see two Icons appearing in the top-right of the field. If you click on the first Icon then the Message field will occupy a much larger area so that you can see the full content of your notes. Before we go to the second Icon go to **General>Frequently Used Text** and create a common Text Message such as "Carry out the following checks.....etc". Having done that go back to this Activity Note and click on the second Icon. A separate screen will appear showing the Frequently Used Text message that you have just created. If you 'double click' on the selected text, to highlight it and click the OK button then the text will be copied to this Message area.

'Save' the record. Now let's have a look at the Event via the 'Activity Calendar'

2. Activity Calendar

Go into **CRM>Activity Calendar** and see what we can do with the Activities in this view. This shows all the activities that are scheduled and allows you to schedule your staff, etc

2.1. Defining the Calendar

Go into **General>Settings>Work Times** and you will see the current Calendar. The Activity Scheduling function uses this calendar to determine each Activities Start and End Date/Time. Amend the calendar if you wish then 'Save' the changes

2.2. Using the Calendar View

Go into **CRM>Activity Calendar** and you will see a base Calendar View. This view allows you to see what Activities are scheduled and what Resources (Employees and/or Assets) are required to perform the required activities. The main screen is split into three sections

- The main panel shows Resources, when and on what Activity they are required.
- The upper-right panel shows a calendar for you to select the displayed time periods

- The lower- right panel shows all Overdue Activities

Main Panel

The Main Panel shows all Activities scheduled for each day displayed. The Activities on display can be viewed:

- For single or multiple Employees and/or Assets
- Grouped by Resource or Date
- In Daily, Workday, Weekly, Monthly, or user-selected time periods
- With a horizontal Timeline

You can create new Activities from within the Calendar by placing the cursor into the calendar under a Resource and click on a start time and then drag the cursor to an end time. Now right click in the enclosed area and select '**New Event**' from the list. The Activity Event creation screen will appear prefilled with the Timeslot's Start and End date and assigned to this Resource. Enter some Notes to describe the Activity then click the '**OK**' button to generate the Activity Event. The generated Event will immediately appear in the Calendar View.

The following options are available in this screen for you to try.

Time Scale: To amend the timeslots of the visible panel right click on the scale down the left-hand side and select the displayed time intervals.

Multiple Resources per Activity: Once again double-click on the Activity to make the Event Details appear then click on the drop-down against field '**Resource**'. You can select as many resources as required to carry out the Activity. Therefore select another Employee to this Activity then click the '**OK**' button. Upon return to the Calendar you will see that the Activity is now assigned to two employees and If you move one Resource's Start and End time for a Activity then the other Resource's timeline will also move.

Event Status: The Event Status is displayed as a coloured 'band' surrounding the Activity. To see the assigned 'Band' colours double click on the Activity and click on the drop-down against field '**Show Time As**' and select a Status.

Event Type: The Activity Event itself can be colour-coded to provide an immediate view of the Event Type of this Activity. To see the assigned 'Event' colours double click on the Activity and click on the drop-down against field '**Label**' and select a Event Type. This will immediately change the colour-code of this Activity to match the Event Type.

Drag and Drop an Activity: You can drag an Activity allocated to a single Resource and drop:

- Against another Resource
- In the same Resource but to another timeslot

Amend Timeline: You can amend the timeline against an Activity by dragging the Start or End boundary and dragging it to the Required Time

Calendar Panel

To the right of the screen you will see a panel that shows a Calendar by month. You can view more months by dragging the Calendar Panel boundary left/right/up/down to let you view the desired months. Within this panel you will see the following:

Active Days: Any Days that contain an Activity will be in **Bold**. Clicking on this day will bring up that day in the Main Panel.

View Days: If you Drag the cursor over a few days then release, the Main Panel will immediately

reflect the selected time-span.

Overdue Activities

At the lower right of the Calendar View you will see a panel that shows all the Activities whose End Date is earlier than the current Date. If you double click on any of these then the calendar will display that date and the selected Activity.

12 11. Point of Sale

Ostendo has a comprehensive Point of Sale system that includes:

- Multi-Site Point of Sale Locations
- Raising Retail Orders with Pickup or Delivery options
- Raising Workshop Orders
- Raising and maintaining Lay-by Orders
- Full interactive data entry options by User
- Comprehensive End Of Day accounting functions
- etc.

12.1 Preparation

You should first prepare for Point of Sale by setting up the following

- Payment Methods
- Payment Method Denominations
- Barcodes
- Default POS Customer
- Usercode Site
- POS Employees
- POS Stations
- POS Rules

1. Payment Methods

You should define the different types of payment methods that you are going to use in POS (Example: Cash, Credit Card, Cheque, etc) along with any rounding factors that may apply. Go into **File>Financial Configuration>Payment Methods** and view what is in the Demo Company. Add your own to the list if you wish. We will address the field '**Rounded in POS**' when we look at the POS Rules setup.

2. Payment Method Denominations

Go into **File>Financial Configuration>Payment Method Denominations** and select '**Cash**' then click on the '**Detail**' view to see the denominations that exist against this payment method. If you created your own Payment Method you may wish to define the denominations for the method. For example:

Name	Value
5 Cents	0.05
10 Cents	0.10
20 Cents	0.20
50 Cents	0.50
1 Dollar	1.00
5 Dollars	5.00
etc	

These denominations may be used to assist in calculating End of Day totals

3. Barcodes

The Point of Sale function accepts Barcodes directly into the Order Line or any of the Search functions. The Barcodes can be linked to:

- Descriptors
- Items
- Item Variations
- Serial Numbers

3.1. Descriptors

To enter Barcodes against Descriptors go into **Inventory>Descriptors** and enter the barcode into the 'Barcode' field

3.2. Items

This is used against Items that don't have sub-level variations (Batch Number, Expiry Date, etc) or you do not wish to differentiate down to that level. To enter Barcodes against these Items go into **Inventory>Items** and enter the barcode into the 'Barcode' field.

3.3. Items with variations

For Items that have variations such as Unit of Measure, Batch Number, Colour, etc. you can enter a barcode specifically related to the combination. To do this go to **Inventory** then, in the drop-down list under **Items** select **Item Barcodes**. Select the Item(s) that have sub-level variants and go to the **'Detail'** view. You can now enter the specific Barcode for the exact combination of variants.

3.4. Serial Numbered Items

For Items that are Serial Number Controlled it is not necessary to enter a bar code as described above. The reason for this is that a Serial Number Barcode is attached to a single Item/Serial Number when it is received into stock. Ostendo therefore knows the sub level variations (if any) that are attached to this specific receipt. Simply scanning the Serial Number will enable Ostendo to recognise the Item and its variants.

You should note that duplicate Serial Numbers within an individual Item Code is not allowed in Ostendo. However you may have the same Serial Number used against another Item Codes or Descriptors. In this instance Ostendo will display both occurrences to the User for selection of the correct one

4. Default POS Customer

If you go into **Sales>Customers** you will see a Customer **'Cash Sales Customer'**. Later, in the POS Rules we will define this Customer as the default Customer to prefill the POS Screen during Order entry. It can be amended in the POS entry screen to suit the specific Customer if required

5. Usercode Site

Whenever a Station is started up it uses the standard Ostendo signon screen. The Usercode used during the signon defines the Company Site at which the Station is situated. Therefore, go into **File>SystemConfiguration>User Security and Options** and click on the **'User Options'** tab. Select **'Company'** from the drop-down against field **'User Site Name'**. We will look at other POS Sites in Exercise 9.4.

6. POS Employees

You need to identify an Employee as a POS Operator and also define other options relating to POS. Go into **Labour>Employees** and select **Bob Drum**. In the **'Detail'** screen complete the following:

- **'Check'** the checkbox to identify that **Bob Drum** is a POS Operator
- Enter a **Password** that will be used when signing into a POS Station
- Select a default **'Sales Person'** that will become the Salesperson for all POS Orders
- **'Check'** the **'Can change POS Prices'** and **'Can change Sales Date'** checkboxes
- In the drop-down against **'Show the following POS related buttons'** select **all** entries. These activities will then be available on the main POS screen and we will go through them in Exercise 9

7. POS Stations

Go into **POS>Settings>POS Stations**. All POS transactions are recorded to the Station where they were transacted. Create a POS Station containing the following:

Computer Name: From the drop-down list locate your computer under **'My Network Places'**

Station ID: Enter (say) **01**

Description: Enter a short description of the Station.

Site Name: For now we are concentrating on a single site POS therefore from the drop-down list select Site **'Company'**

Printer Name: From the drop-down list select the printer that will be used for output of Receipts/Invoices. This is normally a local printer attached to the Station but could be any printer on the network

Print To: For the purpose of these exercises select **'Screen'** from the drop-down list.

Receipt Form Name: From the drop-down list of Reports select **'POS Payment Receipt'**

Invoice Form Name: From the drop-down list of Reports select **'POS Invoice'**

Default Float Value: Enter **'100'** to denote the default float value that is issued to this Station

Default Sales Mode: From the drop-down list select the **'Retail'**. We'll look at the others later.

Cash Drawer ESC: This is the escape command that will trigger the Cash Drawer to open. This would be used with Cash-Drop, etc. For now don't enter anything in this field.

Use Customer Display: If this is 'Checked' then this Station uses an attached Pole Display to show the Customer the current Order total. For the purpose of our exercise leave this 'unchecked'

Customer Display Com Port: Leave this blank. It is the COM Port used by the Customer (Pole) Display.

8. POS Rules

Go into **POS>Settings>POS Rules**. Review the current settings and amend as required. You may wish to use the following settings:

Default Customer Name: **Cash Sales Customer**

Display Line Details: Leave 'unchecked'

Prices Include Tax: 'Check' this checkbox

Round to Nearest: **0.05**

Round up from: **0.03**

Layby Days: **30**

Layby Days Text Code: Leave blank

Default Payment Method: **Cash**

Payment Method used for Cash: **Cash**

9. Cost Centre Mapping for POS

To enable Financial Journals to be created from POS there are two additional Costs Centres that need to be set up. Therefore go into **General>Cost Centres** and create the following

Cost Centre Name: POS - EOD Variance
Description: Point Of Sale End-Of-Day Variance

Cost Centre Name: POS - Shop Expenses
Description: Point Of Sale Shop Expenses

The next step is to go into **File>Financial Configuration>Cost Centre Mapping** and map the above two Cost Centres as follows:

POS End of day Variance: POS - EOD Variance
POS Shop Expenses: POS - Shop Expenses

12.2 POS - Retail Counter Sale

This process Flow takes you through taking a POS order in a **Retail** environment where the goods are presented and paid for at the Counter. (I.e. Using the **Counter Sale** Delivery mode)

Click on **POS>Point Of Sale** and sign in as **Bob Drum** with your password

Let's have a look at the main screen.

- The buttons are all 'Coded' so that you can use the keyboard
- The Buttons down the right-hand side are those you defined against **Bob Drum**
- Immediately under the data entry panel is a thin 'button'. I.e.:
If you click on this button then details of the current line can be viewed. Of course you haven't added a line yet so nothing will be shown. Click on the button against to return to the base view.

You should see that the **'Default Delivery Mode'** is set to **'Counter'**. This is the default for Ostendo. Leave this 'as is' for now. We will look at the other options later.

You will see that the POS Rules **'Default Customer'** populates the **'Billing Customer'** field at the top of the screen. Note:

- If you click on the 'spyglass' symbol in the **'Billing Customer'** field then a separate panel will appear from which you can select a specific Customer currently existing in Ostendo.
- If you click on the **'Customer'** button at the bottom of the screen (or press **F4**) then a separate panel will appear for you to (a) maintain the current Customer, or (b) add a new Customer and then select that Customer from the 'spyglass' in the **'Billing Customer'** field

1. Adding Lines

When you first enter the screen the cursor is positioned in the **'Code'** field. There are three styles of data entry:

- Using Barcode Readers, which scan the barcodes. Note: Duplicate Barcodes could occur for different Items. (Example: You have Supplier's who, for different Items, use the same barcode). If Ostendo encounters this condition then a separate screen is displayed for you to select the specific Item.

- Keying in the Barcode, which functions in a similar way to being scanned by a Barcode Reader.
- Clicking on the 'Spyglass' icon in the Code field and selecting either Item or Descriptor, then locating the required entry

Key in one of the barcodes you created in exercise 1.3. This field will be replaced with the Item or Descriptor Code and the cursor will move to the 'Qty' field

The Qty field is prefilled with **1** and can be amended as required. Upon entry of a quantity the cursor will go to a new line for entry of the next sale line. For the purpose of this exercise highlight the first line in the POS order and then click the 'Edit' button at the bottom of the screen (or key **F9**). This will bring up a panel in which you can edit the details against this line. All fields may not be available depending upon the Employee settings you made in Exercise 1.5. Click on the **Edit** Button (or **F9**) again to return to the default POS view

You will notice that the total POS Order Sell Price appears in the top-right of the screen.

2. Other options

Let's have a look at some other options that are now available

If you click **F5** and then **F3** you have effectively Added a new line and brought up a screen for selection of an item Code. Click the 'Cancel' button then **F7** to remove the line. Finally, add another Line to this Order.

You have the option to place the current Order 'On Hold' while you continue with another order. To do this click on the **F10** button and Ostendo will place the current order 'On Hold'. You can recall it by, once again, clicking on the **F10** button and selecting the Order that was placed 'On Hold'

You also have the option to 'Lock' the workstation while you are away. To do this click on the 'Lock' button (or **F11** key) at the bottom right of the screen. You are required to re-key your username and password to get back into the session.

3. Take Payment

Add another couple of lines to the order and then click on the 'Pay' button (or **F12** key). You will see that the 'Sale Total' field is filled with the value of this sale.

Click on the 'Add' button (or **F5** key) and you will see the default payment method you defined in POS Rules is used. You should also note that the Payment Method 'Cash' has 'Rounding' applied to it. The amount of rounding is defined in **POS Rules**. If your Sales Total contains a value that requires rounding then Ostendo will do this for you and show the rounding amount in field **Rounding**.

You can have multiple payment methods linked to a single Sales Amount.

Upon completion of the payments simply click on the 'Print' button (or **F12** key) and a receipt will be printed. You will be asked if the 'Invoice Printed Correctly'. If you respond with 'No' then you can immediately reprint it. If you respond 'Yes' then the current order will close and the POS will revert to the opening screen in preparation for the next order

12.3 POS - Retail Pickup Order

This process flow takes you through taking a POS order in a **Retail** environment using the **Pickup Default Delivery Mode**. The Pickup option allows you to define selected (or all) lines in a POS Order that are to be collected at a Pickup location. The Point of Sale Receipt is presented at the Pickup location where the collection is confirmed

1. Create Pickup Order

Create a new POS '**Retail**' Order then select '**Pickup**' from the drop-down against field '**Default Delivery Mode**'. Add **100** off Item Code **100-2002** (Washer-Mild Steel-10mm) to the Order and then click on the '**Edit Line**' button (or **F9** key) at the bottom of the screen. The details of the line are displayed where the '**Delivery Mode**' is prefilled with '**Pickup**'. You can amend this to (say) '**Counter**' if this line is not being picked up. Click on the '**Edit Line**' button (or **F9** key) to return to the lines display.

Add another line for (say) **1** off Item **1800-2191** (Front Wheel and Steering Assembly) and leave the '**Delivery Mode**' as '**Pickup**'

Take note of the **Sale Number** in the centre Info Bar. This will be printed on the Payment Receipt and you need to refer to this number when collecting the Goods

Take payment for the order and print the Receipt

2. Confirm Pickup

Go to **POS>POS Pickups** where you will see the current outstanding Pickups. There is one line for each POS Order Line to be picked up. Highlight a line covering the above Sale Number and click on the '**Detail**' tab. 'Check' the '**Goods Picked Up**' checkbox and select the '**Actual Pickup Date**' from the drop-down calendar. Enter any Notes as required and then click the '**Save**' button.

If you go into **Inventory>Inventory Availability**, select the Item and click on the '**Transaction History**' tab you will see that the Item has been issued from the Default Warehouse/Location for the Site

12.4 POS - Retail Delivery Order

This process flow takes you through taking a POS order in a **Retail** environment using the **Delivery Default Delivery Mode**. The Delivery option allows you to define selected (or all) lines in a POS Order, that are to be delivered to the Customer premises.

1. Create Delivery Order

Create a new POS '**Retail**' Order then select '**Delivery**' from the drop-down against field '**Default Delivery Mode**'. Add **1** off Item Code **500-2168** (Sheet-Mild Steel-1200mmx2400mm) to the Order and then click on the '**Edit Line**' button (or **F9** key) at the bottom of the screen. The details of the line are displayed where the '**Delivery Mode**' is prefilled with '**Delivery**'. As with Pickup above you can amend this to another Delivery Mode if required. For now leave the '**Delivery Mode**' as '**Delivery**'

Take note of the **Sale Number** in the centre Info Bar.

Take payment for the order and print the Receipt

2. Deliver the Goods

Go to **POS>POS Deliveries** where you will see the current outstanding Deliveries. There is one line for each POS Order Line to be delivered. Highlight a line covering the above Sale Number and click on the **'Detail'** tab. You can amend the **Planned Delivery Date** if you plan for another date rather than the prefilled system date. You can then schedule deliveries by Date if required.

Once the Goods have been delivered then add a Shipping Reference and 'Check' the **'Goods Delivered'** checkbox and select the **'Actual Delivery Date'** from the drop-down calendar. Enter any Notes as required and then click the **'Save'** button.

If you go into **Inventory>Inventory Availability**, select the Item and click on the **'Transaction History'** tab you will see that the Item has been issued from the Default Warehouse/Location for the Site

12.5 POS - Workshop Order

This process flow takes you through taking a POS order in a **Workshop** Sales Mode. This mode allows you to create, Invoice, and receive payment for simple Workshop Orders

1. Create Workshop Order

To create a Workshop Order select a Customer from the drop-down list under **'Billing Customer'**. You can amend the Customer Details by clicking on the **'Customer'** Button at the bottom of the screen (or the **F4** button). If the Customer doesn't exist then click the **F4** button in the Customer Screen and add the new Customer details. You can then select the created Customer from the drop-down list in the main POS screen.

Now select **'Workshop'** from the drop-down under **'Sales Mode'**. In the POS screen enter:

- **Required Date:** Select today's date
- **Reference:** Enter (say) **'20,000Km Service'**
- **Instructions.** Add a few Instructions relating to the Service

Click on the **'Print'** Button (or **Alt-P**) to print the Worksheet if required. Alternatively you can do this in the next step.

Click on the **'Hold'** button at the bottom of the screen (or key **F10**). This will place the current order on hold until the work has been completed. This facility allows you to continue with other Orders in the POS screen.

2. Carry out the Workshop Order

Go to **POS>POS Workshop** where you will see the current outstanding Workshop Orders. Select the Order you created above and click on the **'Detail'** tab.

You can carry out the following:

- Record the progress of the Workshop Order using the Tracking Code
- Enter Internal Workshop Notes (For example, Items and Labour used)
- Add Details that will appear on the Customer's Invoice

3. Record the Time and Materials

Once the Workshop Order has been completed you should go back to the POS Screen and recall

the Order by clicking the 'Hold' button or pressing the F10 key. From the work carried out in the previous step add the Materials Used and the Labour time consumed (I.e. Use Descriptor 'General Time'). You can place the Workshop Order back on 'Hold' until the Customer arrives and pays for the Order

4. Take Payment

Upon completion of the entries simply click on the 'Print' button (or F12 key) and an Invoice will be printed. You will be asked if the 'Invoice Printed Correctly'. If you respond with 'No' then you can immediately reprint it. If you respond 'Yes' then the current order will close and the POS will revert to the opening screen in preparation for the next order

12.6 POS - Layby Order

This process flow takes you through taking a POS order in a Layby Sales Mode. This mode allows you to create a Layby Order, receive payment, and finally issue the Layby Item(s)

1. Create Layby Order

To create a Layby Order select a Customer from the drop-down list under 'Billing Customer'. You can amend the Customer Details by clicking on the 'Customer' Button at the bottom of the screen (or the F4 button). If the Customer doesn't exist then click the F4 button in the Customer Screen and add the new Customer details. You can then select the created Customer from the drop-down list in the main POS screen.

Now select 'Layby' from the drop-down under 'Sales Mode'. In the POS screen enter:

- **Layby Start:** Prefilled with the system date but you can amend it if required.
- **Layby End:** Uses the 'Layby Days defined in POS Rules to calculate the end date but you can amend it if required.
- **Instructions.** You can add any instructions that relate to this order

Add 1 off Item 2000-2189 (Standard Green Wagon) to the Order. You may find that the sell price in the database is zero. Therefore click on the 'Edit Line' (F9) button and enter a Unit Price of (say) \$200 Click the 'Edit Line' (F9) button again to exit the details. Note: You can change Sell Prices because you defined this in Exercise 1.6

Click on the 'Pay' Button (or F12) and enter an initial Payment of (say) \$25

Click on the 'Print' Button (or Alt-P) to print the payment Receipt. This receipt shows the Layby Item along with the first payment. Confirm that the Receipt has been printed and return to the main POS Screen

2. Receive further Payments

In the POS Screen recall the Order by clicking the 'Hold' button or pressing the F10 key. Click on the 'Pay' button (or press F12) Enter a second payment of (say) \$50 then click on the 'Print' Button (or Alt-P) to print the payment Receipt. You will see a payment history is included in the Receipt.

3. Layby Issue

Whenever the first payment is made against a Layby Order Ostendo Issues the Item(s). You would normally hold these Items in a special Warehouse/Location until payments have been completed.

12.7 Gift Vouchers

This session describes how you can issue Gift Vouchers and subsequently accept those Vouchers as payment against a POS Order.

During this Exercise we will create \$5 and \$10 Vouchers and link these to a Cost Centre. The value will be debited to that Cost Centre when the Voucher is sold. Upon presentation of the Voucher against a POS Order the Voucher's value will be credited to that same Cost Centre.

1. Setting Up

1.1. Defining the Gift Voucher Cost Centre

Go into **General>Cost Centres** and create a new Cost Centre called (say) '**Vouchers**'

1.2. Creating the Voucher(s)

Go into **Inventory>Descriptors** and create a Descriptor with:

Descriptor Code: \$5 Gift Voucher

Unit: Each

Description: Enter your own description as required

Barcode: Enter your own barcode for use in POS Order entry

SP (Inc Tax): Enter 5

Cost Centre: Select **Vouchers** from the drop-down list

Create another Descriptor with:

Descriptor Code: \$10 Gift Voucher

Unit: Each

Description: Enter your own description as required

Barcode: Enter your own barcode for use in POS Order entry

SP (Inc Tax): Enter 10

Cost Centre: Select **Vouchers** from the drop-down list

1.3. Setting up the Payment Account

You should first define the account to which the Gift Voucher will be posted when it is presented as payment against a Sale. Therefore go into **File>Financial Configuration>Payment Accounts** and add the following

Code: Gift Voucher

Description: Enter your own description as required

Cost Centre: Select **Vouchers** from the drop-down list

The next step is to link that account to a Payment Method. Therefore go into **File>Financial Configuration>Payment Methods** and add the following method:

Method: Voucher Receipt

Description: Enter your own description as required

Available in POS: 'Checked'

Rounded in POS: 'Unchecked'

POS Bank Account Code: Select **Gift Voucher** from the drop-down list

2. Selling Gift Vouchers

Create a 'Retail - Counter Sale' POS Order and enter the Gift Voucher barcode into the '**Code**' field, or click on the 'spyglass' icon in this field; click on the '**Descriptor**' button and select the '**\$5 Gift Voucher**'. Enter the quantity then click on the '**Pay**' button (or **F12**) and go through the

payment process using (say) 'Cash' as the payment method

3. Using the Gift Voucher as Payment

Create a 'Retail - Counter Sale' POS Order and enter the product that you are selling (say) '1500-2188' (Yellow Paint). Enter the sale quantity then click on the 'Pay' button (or F12). In the Payment screen click on the 'Add' button then select **Voucher Receipt** as the Payment Method. Enter the Value of the Gift Voucher(s) and, if there is any residual amount, click the 'Add' button and enter another payment Account/Amount for the balance

12.8 Cash Drops

The Cash Drop function enables two key operations to be carried out during the course of the day-to-day activities

- Cash Drop In and Out
- Pay In and Out

1. Cash Drop Out

A 'Cash Drop (Out)' is where monies are taken from the Till. This can occur at any time and the Cash withdrawn is moved to an 'Office Balance' Account relating to this Station. The content of the 'Office Balance' is used during the 'End Of Day' function (see Exercise 8).

To see this in action click on the 'Cash In and Cash Out' button to the right of the screen (or simply click the **Ctrl-6** keys. A panel will appear in which you should highlight 'Cash Drop (Out)' and click the 'OK' button. A panel will appear for you to enter the amount of Cash withdrawn and the reason (if any). Click the 'Accept' button (or key **F12**) to confirm the Cash Drop.

2. Cash Drop In

A 'Cash Drop (In)' is where the Till 'Float' is being 'topped up' from the Office. This can occur at any time and the cash added is moved from an 'Office Balance' Account relating to this Station. As before, the content of the 'Office Balance' is used during the 'End Of Day' function (see Exercise 9).

3. Pay Out

A 'Pay (Out)' is where a cash withdrawal has been made from the Till to purchase miscellaneous items or services. This is effectively treating the Till as 'Petty Cash'. This type of transaction can be made at any time and the amount is treated as a negative payment against the Default Customer defined in the POS Rules using the payment method also defined in the POS Rules

A report is available showing all these 'Petty Cash' transactions

4. Pay In

A 'Pay (In)' is where cash is added to the Till. As before, this is effectively treating the Till as 'Petty Cash' and allows for return of Cash or 'change' from a previous Petty Cash withdrawal. This type of transaction can be made at any time and the amount is treated as a positive payment against the Default Customer defined in the POS Rules using the payment method also defined in the POS Rules

12.9 End of Day

This takes you through the process required either at end of day or whenever an Operator signs off the Workstation.

Go to **POS>POS End Of Day**. Upon entry into the screen Ostendo will calculate the closing balance based on the Opening Balance, Payments, Cash receipts and Issues, and Office Drops. For each Payment Method enter the amount found in the Till. You should note that, for cash, you can click on the icon in the **Closing Balance** field and enter the number of each denomination.

If you have to continue with the Till after entering these figures then you should click on the **Remove End Of Day** button, carry out the transaction, and create a new End Of Day with the revised Till values.

If you click on the **X-Report** Button a report will be produced showing the Opening and Closing balances for the day. This is for information only.

If you click on the **Z-Report** Button then, not only will the report be produced but also all the POS transactions will be 'posted' into Ostendo Journals

12.10 Other Functions

1. Alternate Operator

A Supervisor or other Operator may wish to take over the Station. To show how this is done you should first go into **Labour>Employees** and select (say) **Keith Rogers** and define him as an Operator as described in Exercise 1.6.

Now sign onto POS as **Bob Drum** and create a simple Retail - Counter Order then complete the payment. Now click on the **Operator** button (or **Alt-O**) and sign in as **Keith Rogers**. All transactions are now linked to **Keith Rogers**. This feature is normally associated with a Supervisor (who has Price Override authority) can override a price.

2. 'On Account' Sales

You have the option to raise a POS Order and issue the goods but the payment is posted to the Customer's Account for subsequent payment. To demonstrate this create a POS **Retail - Counter** Order and select Customer **Jim Gold & Co Ltd** from the drop-down list under field **Billing Customer**. In the lines panel add **2** hours of Descriptor **GeneralTime**.

Click on the **Pay** button (or **F12**) and, in the presented panel, you will see a field **Payment Style**. Select **On Account** from the drop-down list. The format of the screen will change and you can enter the Customer's Purchase reference in the **Purchase Ref** field. Note: This is a mandatory entry if the Customer Master record has the **Purchase Order Mandatory** checkbox set to 'True'

Click the **Print** Button (or **Alt-P**) to print the payment Receipt.

If you now go into **Sales>Batch Invoicing** you will see that the Invoice has been created and is awaiting payment.

You may also wish to go to **Sales>Customers>Customer Statistics** and view the history information against the Customer

3. POS History

There may be occasions where you wish to view and reprint Invoices. If you go into **POS>POS History** then this screen will facilitate this.

4. Multiple POS Sites

Up to now you have used a POS application at a single site within the Company. You can create multiple sites and operate the POS sites independently.

4.1. Creating Company Sites

To create a new Company Site go into **General>Company Sites** and add a Site (say) **Retail Store**. For now leave the POS default fields blank

4.2. Multiple Warehouses/Locations

You can have multiple Warehouses and Location per Site. Each Warehouse can have their own Cost Centres related to Ostendo Activities. We will create a Warehouse and Location and then make that the default against the above new Site. Go into **Inventory>Warehouses** and add a new Warehouse (say) **Site1**. You will see that you can assign specific Cost Centres to this Warehouse if Inventory Accounts segregation is to be enforced.

Now go to **Inventory>Locations** and add a single Location (say) **POS** within the above Warehouse

Go back to the New Site and make this Warehouse and Location the default against that record

4.3. User Site link

Go into **File>System Configuration>User Security and Options** and add yourself as a new User. In the '**User Options**' tab select the new Site from the drop-down against field '**User Site Name**'.

4.4. Site Stations

Go into **POS>POS Stations**. All POS transactions are recorded to the Station where they were transacted. Create a POS Station linked to your new Site

4.5. User Signin

Under **File>Switch User** sign in under your new User Name. As the Ostendo User is linked to the new Site then all Operators signing in will also be linked to this site. Therefore click on **POS>Point of Sale** and sign in as **Bob Drum**.

5. POS Support Buttons

The displayed buttons down the right-hand side are specific to the Operator (as defined against the Employee record - see exercise 1.6). Let's have a look at each of these:

5.1. Inventory Availability (or use Ctrl-1)

This will bring up the standard Ostendo Inventory Availability screen so that you may view the

current and projected stock for a selected Item

5.2. Price Inquiry (or use Ctrl-2)

This will bring up the standard Ostendo Price Inquiry screen so that you can determine sell prices 'on the fly' without actually creating a POS Order line.

5.3. Customer Statistics (or use Ctrl-3)

This will bring up the Customer History screen, which shows complete sales and Invoicing history by Order Customer, Billing Customer, or both.

5.4. Sales Orders (or use Ctrl-4)

This will bring up the main Sales Order screen so that you can view all current sales orders in Ostendo

5.5. Linked Images (or use Ctrl-5)

This will bring up a screen showing any Images that are linked to current Item or Descriptor record. Note: If the Item or Descriptor contains Image records then the first image will be displayed in the upper part of the screen (Not valid for screens set to 800 x 600)

5.6. Cash In and Cash Out (or use Ctrl-6)

This allows cash In/Out transactions to be carried out during the current session as described in Exercise 7.

5.7. POS History (or use Ctrl-7)

This will bring up a panel that shows the Receipt history of Sales carried out. This enables any Receipts to be re-printed, if required. Simply highlight the line and click the 'Re-Print Invoice' button

5.8. Cash Drawer (or use Ctrl-8)

Clicking on this button will open the Cash Drawer for miscellaneous transactions such as Cash Drops, etc

5.9. End of Day (or use Ctrl-9)

This will bring up a panel for entry of 'End of Day' balances

13 12. Constraint Based Scheduling

Constraint-based Scheduling uses the fact that every organisation has at least one Resource which can be identified as being a 'Capacity Constraint Resource' (CCR). Any effort in turning that Resource into a non-CCR will only result in other Resources becoming CCR's. It is therefore better to control the current known CCR's and ensure that the maximum throughput is maintained. Synchronised Manufacturing was introduced as the Theory Of Constraints (TOC) by Dr Eliyahu Moshe Goldratt in his book "The Goal" which addresses the control of CCRs in various manufacturing, engineering, and other business processes.

13.1 Terminology

Let us look at the terminology being used:

Labour Code: Within Ostendo a Labour Code can refer to either a Labour Skill (e.g. Machine Operator, Fitter, Electrician, etc) or a piece of Equipment (e.g. Assembly Line, a specific Tool, etc)

Resource: Within a Labour Code you can specify details of what it contains and that includes the quantity of that Labour Code that is available. I.e. The number of 'Resources'.

Capacity: These are the hours that a Labour Code has available to be used. It is expressed as the hours available per day * the quantity of Resources in this Labour Code.

Load: This is the quantity (in hours) that the Labour Code is required and is taken from the lines in the Job or Assembly order

Finite Capacity: If the Labour Code is defined as being subject to "Finite Capacity" planning then the Capacity cannot be exceeded when applying the Load

Infinite Capacity: If the Labour Code is defined as being subject to "Infinite Capacity" planning then the Capacity is NOT a factor when applying the Load. However it does enable you to compare the evaluated Load to the available Capacity

13.2 Brief Overview of the Process

Ostendo's Constraint-based Resource Scheduling routine starts by prioritising the Job Orders and/or Assembly Orders. This can be based on combinations of up to 3 of the following:

Order Sequence - A manually entered Order or Product sequence number. This is useful for scheduling a flow-line where the product has slight variations. For example: A Dye Vat where the Vat is loaded with light colours first and progresses through to dark colours before the vat has to be cleaned.

Required Date - The Order Required Date sequence

Order Date - The Order Creation Date sequence

Critical Ratio - The Order's Critical Ratio which is an index number that Ostendo computes by dividing the time remaining to the Order's due date by the work time remaining

Order Priority - A user-defined Priority Number held against the Order

Order Started - If the Order has already started then this is prioritised before Orders that have not yet started

Having Prioritised the Orders Ostendo will then look at the Resource Requirements within each order in turn and load the Resource in accordance with any Capacity Constraints defined against the Labour Code.

Ostendo will then produce a recommended 'Work-To' List and Gantt Chart. Other Views, Reports and Graphical presentations are available to support this output

Although the scheduling process takes into account Capacity, Order Priority, and Resource

availability there are situations where, Inventory or Resource shortages still occur; For example - the buffer stock was found to be faulty or, the Employee has gone off sick. The Production Planner therefore has the option to adjust this Lists or Gantt Charts to take account of short-term changes such as Urgent Orders, Scrapped Inventory, etc).

In this series of exercises we start by setting up a simple introduction to Constraint-Based Resource Scheduling in which we will:

- Create a single Labour Code and define its Scheduling parameters.
- Create a single Item and its associated Bill of Material containing this single Resource requirement
- Create a single Assembly Order for this Item
- Run the Resource Scheduling routine and look at the results along with associated screen and views

We will then carry out further examples to look at:

- Loading Options: Infinite Capacity, Finite Capacity, Finite Capacity with alternate Resources
- Buffers, Overlaps and Parallel Steps/Tasks
- Minimum Gaps and Continuous Operations
- Planned and Firm Schedules

13.3 The First Exercise

1. Creating a Labour Code

In Ostendo's Constraint-based Scheduling a Labour Code does NOT refer to a specific Employee but rather a Skill Level, a piece of Equipment, or anything else where you require a detailed analysis of throughput loading.

Obviously you should first define the Labour Code. Let us begin by creating a Resource. Go to **Labour>Labour Codes** and create a new code:

Labour Code:	RES-CRIT
Unit:	Hours
Description:	Critical Resource
Department:	Select 'Production' from the drop-down list

Having defined the Labour Code we now need to tell Ostendo if it is to be scheduled; plus how (and if) it is constrained. Therefore, in the Labour Code's 'Detail' screen 'check' the 'This Code is Scheduled' checkbox. Note: The 'Daily Capacity in Hours' field in this screen is NOT used in Scheduling as the capacity is evaluated from information entered into the 'Scheduling Information' screen. To get to this screen click on the 'Scheduling Information' Button. Let us look at the available options.

- **Always Swap with Code:** You could have many BOM Routings that nominate a Labour Code that is temporarily out of service (Example: Equipment that is being repaired). Rather than amend every BOM so that it refers to the new Labour Code you can simply select another Labour Code here. This will take the place of the 'old' Labour Code during Scheduling. We will cover this later in these exercises
- **Capacity Type:** There are two types:
 - Infinite** - Resource Scheduling routine will not take any Capacity into account when loading the activities. However you can compare the required load (obtained from Job and/or Assembly Order Lines) against the available capacity identified here
 - Finite** - Resource Scheduling routine will take the Capacity into account when loading the activities. I.e. It will NOT load the Resource to more than its available

Capacity

- **Continuous Operation:** Some operations require continuous running – typically in the process industries or where setup time is large. If you 'check' this checkbox then the activity will only be loaded if there enough time to complete it continuously.
- **Efficiency %:** Where a Labour Codes or group of Resources is not performing to their standard rates it might be necessary to adjust the efficiency, rather than adjusting all the bills of material or job templates. If this is the case then you should enter the efficiency that you wish to apply (Example 50% efficient means the activity will takes twice as long as the base time)
- **Number of Resources:** You should define the quantity of this Resource that is available for Scheduling. Example: If you have a machine shop Labour Code with 4 centre-lathes that all perform the same activities then 4 should be entered here
- **Minimum Gap in Minutes:** Where the Resource is NOT flagged as Continuous Operation you can still define a minimum amount of time that must be available before this Resource can be scheduled.
- **(Start and End Times):** You can define the Start and End Times for each day of the week that this Resource is available.

For the created Labour Code **RES-CRIT** enter the following

Labour Code:	RES-CRIT
Unit:	Hours
Description:	Critical Resource
Always Swap with Code:	(Leave blank)
Capacity Type:	Infinite
Continuous Operation:	Leave this 'unchecked'
Efficiency %:	100
Number of Resources:	2
Minimum Gap in Minutes:	(Leave blank)
(Start and End Times):	Enter Start 08:00:00 and End 17:00:00 against Monday to Friday

Save the settings

The above entry means that Resource **RES-CRIT** has a weekly capacity based on:

<u>Day</u>	<u>Hours in use</u>	<u>No of Resources</u>	<u>Total Capacity (hrs)</u>
Mon	9	2	18
Tues	9	2	18
Wed	9	2	18
Thur	9	2	18
Fri	9	2	18

NOTE: For the purpose of this Training exercise you should go into the other Labour Codes and 'uncheck' the '**This Code is Scheduled**' checkbox. This will ensure that Resource Scheduling will focus on **RES-CRIT** Labour Code only. **Hint:** Go to the **Labour Resource - List** screen and 'right mouse' in the centre of the screen. Click on '**Customize Fields**' and select '**RequiresScheduling**'. You can now see which Labour Codes are currently scheduled

We will now create an Assembled Item that uses this Labour Code

2. Creating the Assembled Item

Go to **Inventory>Items** and create a new Item called (say) '**PRODUCT-A**' with the following fields completed.

Item Code:	PRODUCT-A
Unit:	Each

Description: Product to show Resource Scheduling
Sourced By: Assembly

Also create a new Item called (say) 'RAW MATERIAL' with the following fields completed.

Item Code: RAW MATERIAL
Unit: Each
Description: Raw Material to make PRODUCT-A
Sourced By: Purchased

The next step is to create a Bill of Material so go to **Assembly>Bills of Material** and create a new BOM for **PRODUCT-A**. Click on the 'Lines' tab and add the following two lines.

Run or Setup: Run
Sequence: 10
Step: Assembly
Line Number: 10
Line Type: Item Code
Code: RAW MATERIAL
Qty: 1

Run or Setup: Run
Sequence: 10
Step: Assembly
Line Number: 20
Line Type: Labour Code
Code: RES-CRIT
Qty: 20

2.1. Create the Assembly Order

Now create an Assembly Order using this BOM. Go to **Assembly>Assembly Orders**. Click the 'Add' button and create an Assembly Order for 1 off Item **PRODUCT-A**.

2.2. Constraint Scheduling.

We will now run the Constraint Scheduling routine. Therefore go to **Requirements>Constraint Scheduling** and select the 'Create New Schedule' Radio Button. 'Check' the Order Classes 'Assembly Orders' checkbox and amend the 'Start Scheduling from' date to today's date. Click the 'OK' button.

Ostendo will return an initial screen showing all orders loaded in the priority sequence identified in the Constraint Scheduling Parameter panel. In this screen you can identify those Assembly Orders that wish to exclude from the Resource Scheduling step by selecting the 'Exclude Order' checkbox against the relevant Order(s). For the purpose of this exercise leave these 'as is'. If you now click on the 'Continue' button at the bottom-right of the screen Ostendo will, for the orders not excluded, assess the Resource Loading requirements in order priority sequence and display the results in the 'Scheduling Board' screen. As you have told Ostendo to ignore all other Labour Resources (see NOTE in para 2.1 - above) then this screen should only show the **RES-CRIT** Resource. You will see that Ostendo has loaded **RES-CRIT** to the available total capacity (18 Hours) starting at the beginning of the day defined by the 'Start Scheduling From' date. The balance of the load (2 Hours) for **RES-CRIT** is now carried over and loaded into the next day.

You will notice that the even though the Capacity Type is 'Infinite' the Load uses the available Capacity for the first day before proceeding to the next day for the balance. We will expand on this in the next Section covering Capacity Types

Let's look at what else is available in the 'Scheduling Board' screen.

Click on the '[Graphical Schedule](#)' tab. If you select the Date from the Calendar displayed down the right of the screen you will see the Jobs that are required to be done on this day by the selected Resource

Click on the '[Loading and Capacity](#)' tab. If you select Department '[Production](#)' and Resource '[RES-CRIT](#)' you will see the Load -v- Capacity

Go back to the '[Schedule](#)' tab and click on the '[Reports](#)' button. Run the two reports and view the contents.

13.4 Capacity Types

Let us now look at the various Capacity Types

1. Infinite Capacity

Ostendo uses the current Capacity when loading a single Order. This capacity can be exceeded if further Orders are added that use this same Labour Code. Let's see what happens when we create a second Assembly Order which uses the same Labour Code. Therefore create another Assembly Order for 1 off Item [PRODUCT-A](#) with the Start Date one day later than the above Order. If you re-run the Constraint Scheduling you will see that the second Order will be similarly loaded and the Capacity will be ignored

2. Finite Capacity

For Finite Capacity the Load NEVER exceeds the Capacity. The simplest way to demonstrate this is to go back into [Labour>Labour Codes](#) and - for Labour Code '[RES-CRIT](#)' - click on the '[Scheduling Information](#)' Button. In this panel amend the [Capacity Type](#) to '[Finite](#)'.

If you now re-run the Constraint Scheduling as described earlier you will see that the Loading will NOT exceed the Capacity.

What you may not have considered is that there are two ways in which Load can be applied.

- Apply across all Resources within the Labour Code
- Apply to Individual Resource within the Labour Code

Let's look at each option

2.1. Apply across all Resources within the Labour Code

An example of where this is used could be when there are 3 Assembly Operators within a Labour Code and each Order's load can be spread across all the Operators.

In our example (above) there are 2 Resource '[RES-CRIT](#)' and any Load is allocated across both Resources

2.2. Apply to Individual Resource within the Labour Code

The previous method is OK where the load can be spread across all the available Resources within the Labour Code. However, where the Order requires a large and expensive Setup process before production can begin then the Loading is usually confined to one specific Resource within the Labour Code.

Ostendo deals with this by having a primary Labour Code with a Resource quantity of 1 and linking this to an alternative Labour Code which also has a quantity of 1. During Scheduling if the first

Labour Code is fully loaded then Ostendo will load the alternative Labour Code...and so on. Let's see how this works.

Go to **Labour>Labour Codes** and create another Resource with the following:

Labour Code: RES-CRIT-1
 Unit: Hours
 Description: Critical Resource
 Department: Select 'Production' from the drop-down list
 This Code is Scheduled 'Check' this checkbox.

Click on the 'Scheduling Information' button and enter the following:

Capacity Type: Finite
 Continuous Operation: Leave this 'unchecked'
 Efficiency %: 100
 Number of Resources: 1
 Minimum Gap in Minutes: (Leave blank)
 (Start and End Times): Enter Start 08:00:00 and End 17:00:00 against Monday to Friday

Now, recall Labour Code 'RES-CRIT'. Click on the 'Scheduling Information' button and amend the 'Number of Resources' to 1. While you still in the 'RES-CRIT' record click on the 'Related' button down the right of the screen and select 'Labour Alternatives'. In the presented screen highlight 'RES-CRIT' and click on the 'Detail' tab and add a record with the following information:

Alternate Code: Select RES-CRIT-1 from the dropdown list
 Speed Percent: Leave at 100
 Load Balance for Schedule: 'Check' this checkbox

If you now re-run the Constraint Scheduling as described earlier you will see that the Loading consumes RES-CRIT first and then consumes RES-CRIT-1 over the same timescale

13.5 Scheduling Options

The following options can be specified to adjust the Scheduling to suit your specific requirements.

1. Efficiency

In the above examples we have defined the Resource as being 100% efficient and would, therefore use the standard time entered against the Order. In the previous example the Resource 'RES-CRIT' has an efficiency of 100%. If you go into **Labour>Labour Codes** and - for Labour Code 'RES-CRIT' - amend the Efficiency to 50%. This means that the scheduled 20 hour process will take this Resource (20 * 100 / 50) 40 hours to complete

If you now re-run the Constraint Scheduling you will see that the Loading against RES-CRIT has been increased to take account of the efficiency

2. Fix Schedule

This option allows you to 'Fix' a schedule that you previously generated against an Order. This means that the scheduling process will pre-load these Orders into their current timeslots BEFORE continuing with new Orders.

To identify that an Order's schedule is to be fixed go to **Requirements>Constraint Scheduling** and select the 'Open Last Planned Schedule' Radio Button. This will display the Scheduling Board and show the current Orders. If you scroll to the right then you can 'check' the checkbox in field 'Schedule Fixed'. The dates against this Step/Task will NOT change when you re-run the Constraint Scheduling process.

You may also 'Fix' the Schedule for the whole order in these places:

Assembly Orders: *Assembly>Assembly Orders*. In the 'Detail' screen of the order click on the 'Scheduling' tab and 'check' the checkbox labelled 'The Scheduled Time Has Been Fixed'

Job Orders: *Jobs>Job Orders*. In the 'Detail' screen of the order click on the 'Scheduling' tab and 'check' the checkbox labelled 'The Scheduled Time Has Been Fixed'

3. No Scheduling

You may have some Resources that you do now wish to schedule. For these Resources simply go to *Labour>Labour Codes*. In the 'Detail' screen of the Labour Code 'uncheck' the 'This Code is Scheduled' checkbox. This Labour Code will then be excluded from the scheduling process.

4. Buffers

Buffers are used where there is more than one Step or Task and the following Step/Task has a 'Wait Period' after the current Step/Task has been completed. A good example of this is where Task 1 is 'Lay Concrete' and Task 2 is 'Fit Structures'. This would require a 3-day waiting period on completion of Task 1 - for the concrete to dry - before Task 2 can begin.

Let's see this in action on the BOM that you created above. Go to *Assembly>Bills of Material* and recall **PRODUCT-A**. In the 'Detail' screen click on the 'Routing Steps' tab. You will see that this BOM uses the default 'Assembly' Step. If you scroll to the right you will see two fields that are relevant to setting the Buffer

Step Overlap select 'End of Step' from drop-down list
Buffer Minutes enter 1000

This denotes that the next Step will commence 1000 minutes after completion of this Step

Add another Step using 'QA' Step.

Finally, click on the 'Lines' tab and add the following line.

Run or Setup:	Run
Sequence:	20
Step:	QA
Line Number:	10
Line Type:	Labour Code
Code:	RES-CRIT
Qty:	6

To start with a 'clean slate' when Scheduling go into *Assembly>Assembly Orders* and delete the two Orders you created earlier. Now click the 'Add' button and create a new Assembly Order for 1 off Item **PRODUCT-A**. If you Re-Run the Resource Scheduler and view the results. You will see that Step QA commences after the end of Step Assembly plus 1000 minutes

Note: Buffers are specified in the following places

- *Assembly>Bills Of Material* and in the 'Detail' screen click on the 'Routing Steps' tab
- *Assembly>Assembly Orders* and in the 'Detail' screen click on the 'Routing Steps' tab
- *Jobs>Job Orders* and in the 'Detail' screen click on the 'Job Tasks' button. The Buffer information fields are located at the bottom of the screen
- *Jobs>Templates* and click on the 'Tasks' tab. The Buffer information fields are located at the bottom of the screen

4. Overlaps

Overlaps are where Steps or Tasks can commence before the previous Step/Task has been completed.

Once again start with a 'clean slate' by deleting the Assembly Order you have just created. Now go into the Bill of Material to **Assembly>Bills of Material** and recall **PRODUCT-A**. In the 'Detail' screen click on the 'Routing Steps' tab. You will see the two steps that you created in 2 above. Against the first (**Assembly**) step enter the following

Step Overlap	select 'Start of Step' from drop-down list
Overlap Minutes	enter 60
Buffer Minutes	Blank

This denotes that the next Step (**QA**) will commence 60 minutes after this Step has started.

Now click the 'Add' button and create a new Assembly Order for 1 off Item **PRODUCT-A**. If you Re-Run the Resource Scheduler and view the results. You will see that Step **QA** commences 1 hour after Step **Assembly** has started

5. Parallel Steps/Tasks

Parallel Steps/Tasks are where two or more steps can be carried out at the same time. As before start with a 'clean slate' by deleting the Assembly Order you have just created. Now go into the Bill of Material to **Assembly>Bills of Material** and recall **PRODUCT-A**. In the 'Detail' screen click on the 'Routing Steps' tab. You will see the two steps that you created in 2 above. Against the first (**Assembly**) step enter the following

Step Overlap	select 'Start of Step' from drop-down list
Overlap Minutes	enter 0
Buffer Minutes	Blank

Now click the 'Add' button and create a new Assembly Order for 1 off Item **PRODUCT-A**. If you Re-Run the Resource Scheduler and view the results. You will see that Step **QA** commences at the same time as Step **Assembly**

6. Minimum Gaps

During the Scheduling process Ostendo will attempt to insert the Load into any available Capacity. The situation COULD arise where the current loading shows one or more gaps where there is no current Loading requirement. Obviously this gap may be too small to commence, or restart, working on an Order as this could produce a Schedule where this Order is repeatedly put to one side so that other Orders can be completed...hence the need for Minimum Gaps

You can therefore define a 'Minimum Gap' against the Labour Code. This will inform the Scheduling process that if there are gaps in the Loading then only Load into that gap if it is greater than a specified number of minutes

To set this gap go into **Labour>Labour Codes** and - for Labour Code '**RES-CRIT**' - click on the 'Scheduling Information' Button. In this panel enter the Minimum gap in minutes.

7. Continuous Operations

Some operations require continuous running – typically in the process industries or where setup time is large. If the Resource is defined as is '**Continuous Operation**' then the activity will only be loaded if there enough time to complete it continuously. To define the Resource as Continuous Operation go into **Labour>Labour Codes** and click on the 'Scheduling Information' Button. In this panel simply 'check' the 'Continuous Operation' checkbox. You should note that this setting is used in combination with field '**Number Of Resources**' where the quantity would normally be set to

1

13.6 Planned -v- Firm Schedule

In the above exercises you have created and manipulated the schedule. Once you are satisfied with the finished schedule you obviously would like this to remain 'as is' until it is overwritten BUT at the same time be able to see the effect of any additions and short-term adjustments. You can achieve this in the following manner.

Go back to *Requirements>Constraint Scheduling* and select the 'Open Last Planned Schedule' Radio Button then hit the 'OK' Button. The schedule that you see is the one that you been working with. If you wish this to retain this schedule then click on the 'Firm' button down the right of the screen. This will take a complete copy of the 'Planned' Schedule. After creating the Firm copy exit to the main Ostendo screen.

If you now recall to *Requirements>Constraint Scheduling* you can select either the 'Open Last Planned Schedule' or 'Open Last Firm Schedule' Radio Button then hit the 'OK' Button. Although you can make changes to either of these it is recommended that you only carry out changes to the **Planned Schedule**

14 13. Workflow Graphical Designer

Workflows are user-defined graphical representations of Operational Activities, Training Guides, Installations Guides, KPIs, etc. The Workflow itself can include interaction with Ostendo such that clicking on a Graphical object takes the User directly to a Screen, Report, or View in Ostendo. Additionally, the object can be linked to Custom Menu Scripts that can perform Customised Activities. These activities can be refreshed automatically or each time the object is selected

14.1 Getting Started

Start up Ostendo Graphical Designer. You will find this under the Ostendo Folder as ostdesigner.exe.

Select **'New'** found under **'File'** on the top toolbar.


A blank Canvas will appear. Click on **File>Properties** and set the palette dimensions to **1040 x 600**. (This is the aspect ratio and size that fits within the main Ostendo Desktop)

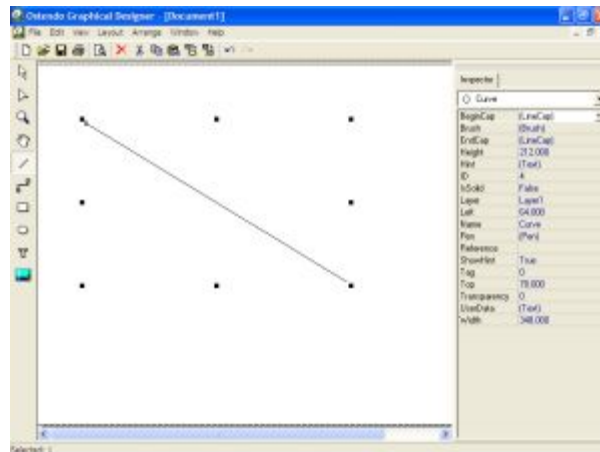
Go to **'View'** on the top toolbar and uncheck every selection except **'Draw Tools'**, **'File Tools'**, and **'Inspector'**. We will bring in the other selections as we progress through this tutorial.

14.2 Graphics Tools

We will start off by creating and manipulating various graphic and text objects


14.2.1 Straight Line

Click on the **'Line Tool'** icon  on the left of the screen. Move the cursor into the canvas then click the left mouse and drag to create a line. The Line tool will remain active and you can create more lines by simply releasing the left mouse button and repeat the function.



When creating each line you can see eight black squares (known as 'handles'). Selecting a handle and dragging it to the required position will move the line.

The area enclosed by the squares is called the **'Object'** and this is used extensively throughout the Graphics function

To exit the '**Line Tool**' click on the '**Pick Tool**' icon.  To focus back onto the line, simply click on the line in the canvas.

You can also amend the Line's colour, style and thickness by using the following selection from the '**Inspector**' functions. The Inspector panel can be seen to the right of the palette. In the list select:

- **Pen** – This allows you to amend the format, width and colour of the line

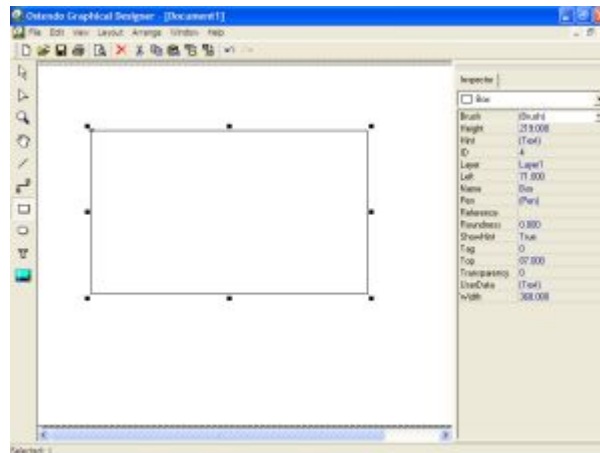
Having selected the Object you can delete it by clicking on the '**Delete**' icon on the top toolbar.

For the purpose of this tutorial, remove the line from the canvas.

14.2.2 Rectangle

Click on the '**Rectangle Tool**' icon  on the left of the screen.

Move the cursor into the canvas then click the left mouse and drag it to create a rectangle. Once again the Rectangle tool will remain active and you can create more rectangles as required.



You should also note that the rectangle is contained within an 'Object' giving you the flexibility to resize the rectangle by dragging its handles.

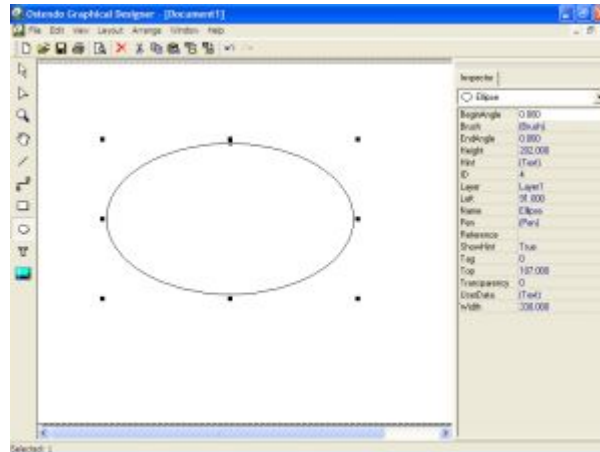
Have a try at amending the 'look and feel' of the Rectangle by using the following selections from the 'Inspector' panel

- **Brush** – This allows you to 'fill' the rectangle with colour. Try these two options.
 - Click on 'Standard' radio button then select a brush style and a brush colour, then hit the OK button
 - Click on the 'Gradient' radio button and select 'Vertical' with (say) colours 'Yellow' and 'Blue', then hit the OK button
- **Size** – If you enter a 'Height' and 'Width' then this will immediately be reflected in the size of the rectangle
- **Position** – If you enter a 'Left' and 'Top' dimension then will immediately position the rectangle in the canvas
- **Pen** – As with the straight Line above this defines the format, width and colour of the border
- **Roundness** – If you enter (say) 30 then the corners of the Rectangle will be rounded.

For the purpose of this tutorial, remove the Rectangle from the canvas

14.2.3 Ellipse

Click on the '**Ellipse Tool**' icon  on the left of the screen.




This now functions in exactly the same way as a rectangle Object.

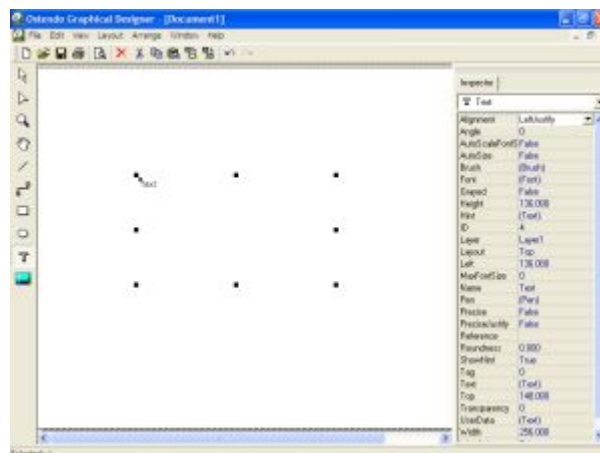
You may wish to also amend the Ellipse by using the following selection from the '**Inspector**' panel

- **Brush** - Click on the drop-down will let you define the fill value for enclosed area then click on the '**Gradient**' radio button and select '**Elliptic**' with (say) colours '**Green** and '**Blue**', then hit the **OK** button

For the purpose of this tutorial, remove the Ellipse from the canvas

14.2.4 Text

Click on the '**Text Tool**' icon  on the left of the screen and create an Object on the canvas.



You will see that the Object contains the text '**Text**'. On the '**Inspector**' panel on the right of the canvas click on '**Text**' and select the drop-down. A panel will appear for you to enter your text.


Type a short text into this panel and click the 'OK' button.

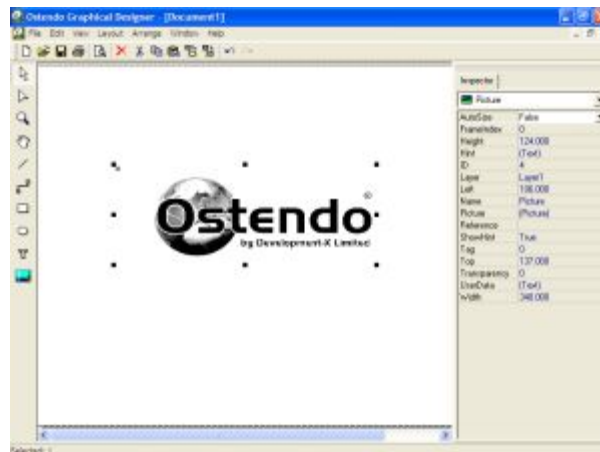
You can also amend the 'look and feel' of the Text by using the following selections from the 'Inspector' functions.

- **AutoScaleFonts** - Automatically sizes the Font in the text box
- **Alignment** - Adjusts the horizontal position of the Text within the Object
- **Layout** - Adjusts the vertical position of the Text within the Object
- **Angle** - This defines the angle (0° to 360°) at which the Text will be displayed
- **Autosize** - Clicking on this will resize the Text box to suit the current Text
- **Brush** - Clicking on the drop-down will let you define the fill value for Object background.
- **Font** - A panel will appear for you to format the Font, Font Style, Size, Colour.
- **Wordwrap** - If this is set to 'True' then the text will be contained within the Text rectangle.

For the purpose of this tutorial, remove all Objects from the canvas.

14.2.5 Picture

Click on the 'Picture Tool' icon  on the left of the screen and create an Object on the canvas.



Drag the mouse to 'size' the picture object. 'Double Click' on the object, or go to the 'Inspector' panel on the right of the canvas and select the drop-down against 'Picture'. A panel will appear for you to locate the picture. Click on 'Load' and find the picture on your computer, then hit the OK button. In this panel you may also check the 'Store as Link' link checkbox and point the link to a location in your database. The picture currently at that location will appear on the displayed document.

Save this canvas by selecting 'Save' found under 'File' on the top toolbar. We will return to this later in this tutorial.


14.3 Curve Edit Tools

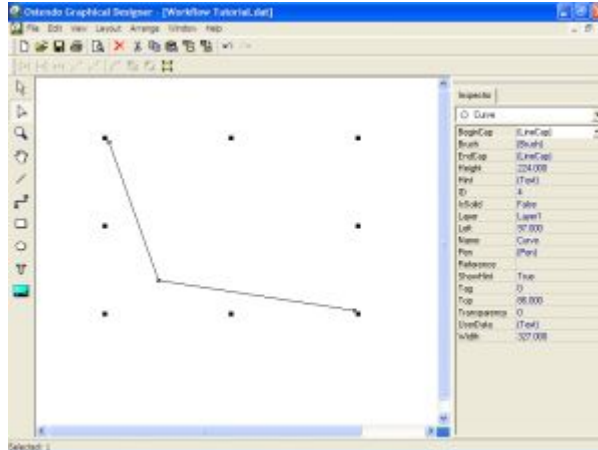
Curve Edit Tools allow you to create and manipulate lines to form any curve.


You should first go to 'View' on the top toolbar and 'check' 'Curve Edit Tools'.

14.3.1 Creating a Triangle

Start by selecting the Line Tool on the 'Draw Tools' Toolbar and creating a line object on the canvas

- Now select the 'Curve Edit Tool'  on the 'Draw Tools' Toolbar.
- Position the cursor close to the line and press the Ctrl key on the keyboard. Dotted lines should appear that connects the cursor position to each end of the line.
- Having positioned the cursor press the left mouse to split and re-position the current line.




- Click on the small white square at one 'open' end of the line and then, holding down the 'Shift' key, click on the small white square at the other end of the line. (Both squares will now be black to denote that they have been selected).
- Now select the 'Close Figure' icon  on the 'Curve Edit Toolbar' and the program will connect the two ends.
- You can now use the Inspector Options for the triangle as if it was a Rectangle

Remove the triangle from the canvas to make room for creating a curve.

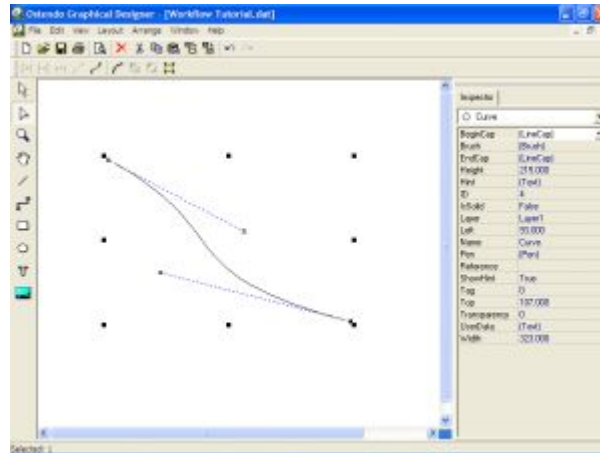
14.3.2 Creating a Curve

Start by selecting the Line Tool on the 'Draw Tools' Toolbar and creating a line object on the canvas

Now select the 'Curve Edit Tool' on the 'Draw Tools' Toolbar. Click on the small white square at one end of the straight line and then, holding down the 'Shift' key, click on the small white square at the other end of the line. (Both squares will now be black to denote that they have been selected).

Select the 'Convert selected segments to curve' icon  on the 'Curve Edit Toolbar'. Although nothing appears to happen you should note that (a) the program will change the squares back to white and (b) the line is now, in effect, a curve.


To demonstrate that it is a curve, select one of the white squares; and two 'nodes' will appear. Select a node and drag it away from its current position. You will see that it forms a curve.

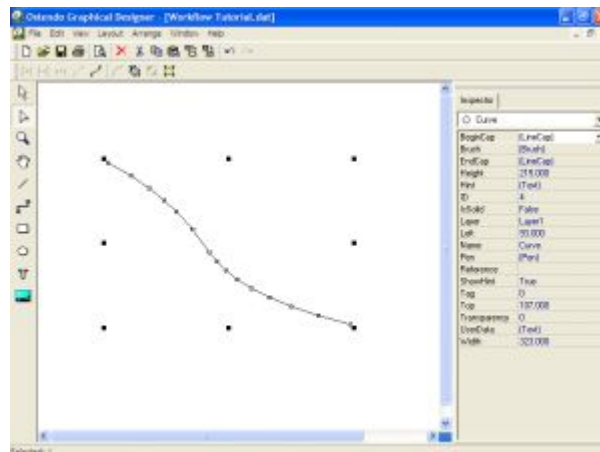



You can manipulate the curve by:-

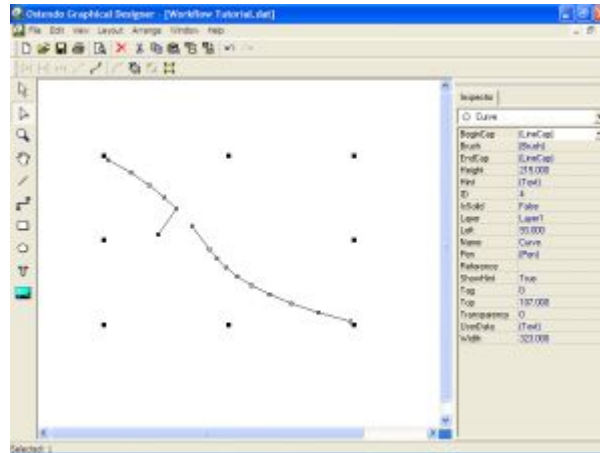
- Dragging the squares at the end of the line
- Dragging the Object's boundary handles
- Dragging the extended 'Nodes' linked to the end squares

14.3.3 Curve Manipulation


Having created the curve in the previous step you should now click on the '**Flatten Selected Curve**'  on the '**Curve Edit Toolbar**'. You will see that the curve will be split into multiple nodes containing straight lines that simulates the curve. You should note that each of the lines between the nodes can now be converted into a curve as described above.




If you click on one of the nodes in the centre of the curve and select the '**Break Selected Nodes**'  button then the curve will, effectively be split. To create the split, click on the node and drag it from its current position.

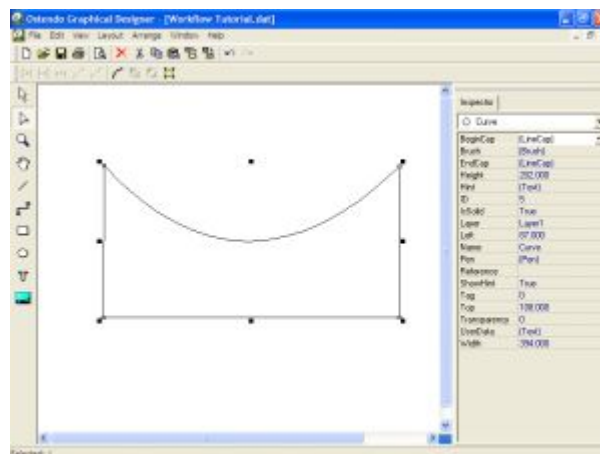


14.3.4 More Curve Manipulation

Place a Rectangle onto the canvas then select the **'Curve Edit Tool'**. In this instance you will find that you cannot carry out the actions as described in the previous examples. This is because the program does not know which line you are addressing. Therefore select **'Convert Selected Objects to Curve Objects'**  icon on the **'Curve Edit Tools'** toolbar. The four corners of the Rectangle will now have the small white squares.

Identify the two small squares in the Rectangle that you wish to convert to a curve. Click on one of the squares and then, holding down the **'Shift'** key, click on the small white square at the other end of the line. (Both squares will now be black to denote that they have been selected).

Select the **'Convert selected segments to curve'** icon  on the **'Curve Edit Toolbar'**. Although nothing appears to happen you should note that (a) the program will change the squares back to white and (b) the line is now, in effect, a curve. Click anywhere in the line and rag it away from its current position.



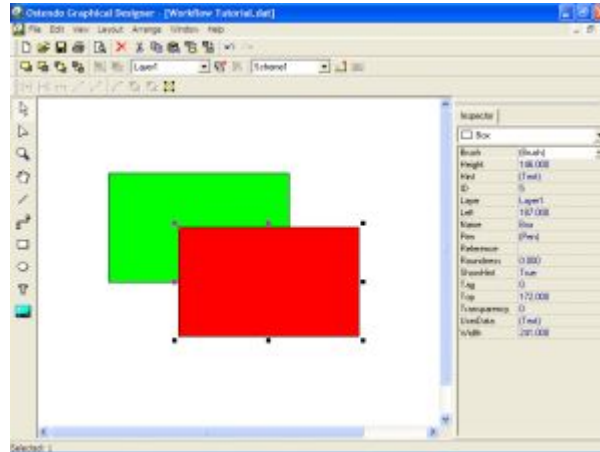
At the end of this exercise remove the object(s) from the canvas in preparation for the next exercise

14.4 Layout Tools





Layout Tools allow you to maintain a relationship between Objects. Go to '**View**' on the top toolbar and 'check' the '**Layout Tools**'.

14.4.1 Viewing priority

The viewing priority defines which object takes priority when viewing the canvas. To demonstrate this create two Rectangle Objects and 'colour fill' one Green and the other Red (Hint:- Use Standard properties in 'Brush' on the 'Inspector' toolbar on the right of the screen). Alternatively you may wish to click on 'View' on the top toolbar and 'check' the 'Palette' Toolbar.

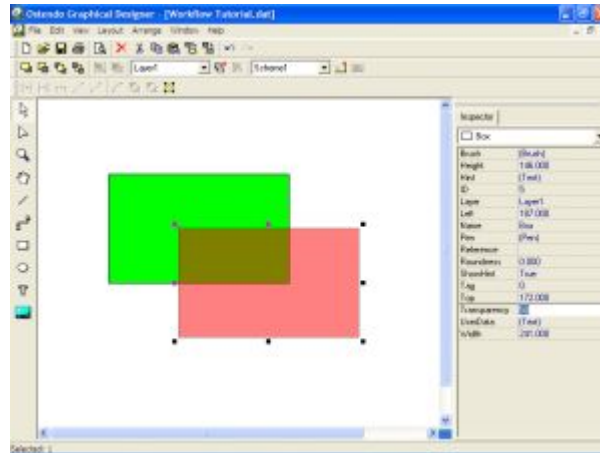


You will note that one coloured rectangle takes preference. To reverse this, click on the front Rectangle then select '**Back One**' on the **Layout Toolbar**. This manipulation can be one of :-

- | | | |
|---|-------------|---|
|  | Forward One | The current Object is brought forward one step |
|  | Back One | The current Object is sent back one step |
|  | To Front | The current Object is brought in front of all Objects |
|  | To Back | The current Object is sent to the back of all Objects |


14.4.2 Transparency

Rather than having two solid colours you can also elect to use Transparency levels. Select the front object that you created above and enter (say) 50 in the '**Transparency**' field of the Object's Inspector. If you exit the field then the object will become semi-transparent. Try altering the Transparency number (0 to 100)

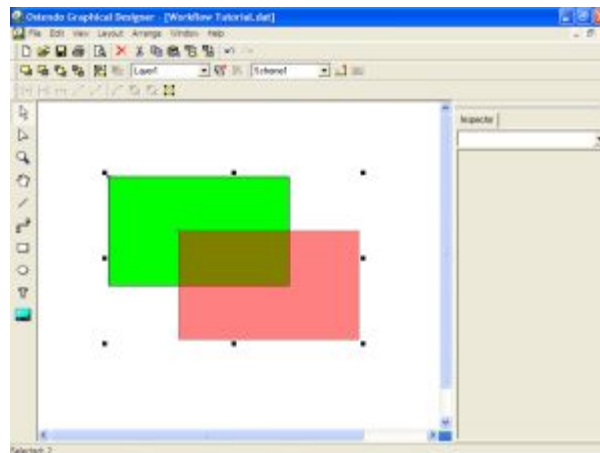



14.4.3 Object Grouping

Of course, in some instances you may wish to combine and then view the relationship between two or more Objects. Taking the two rectangles you created above; you are still able to manipulate them individually. In some instances this is not desirable and they should be combined and moved 'as one'. To combine them first select the **'Pick Tool'** icon on the **'Draw Toolbar'** then - on the canvas - drag the mouse to create a rectangle enclosing the two objects.

Now select the **'Group'** icon  on the **'Layout Toolbar'**. The rectangles will now exist as one 'Group' Object and any changes will apply to both rectangles.

Note:- As an alternative to 'surrounding' the objects you can select all objects to grouped by pressing the 'Shift' key and clicking on each object.



You can ungroup the Objects by selecting the Group Object on the canvas then selecting the **'Ungroup'** icon  on the **'Layout Toolbar'**. The rectangles will now exist as separate Objects.

14.5 Zoom Tools

The Zoom Toolbar enables you to zoom in or zoom out on the canvas. Go to **'View'** on the top toolbar and 'check' the **'Zoom Tools'**.


Try the following on the current canvas:-


- **Scale** - Select a scale from the drop-down list or key in the desired scale. The canvas will zoom about its current viewable centre point.
- **Zoom In** - The canvas will zoom in about its current viewable centre point by a factor of 2 for each click of the mouse.
- **Zoom Out** - The canvas will zoom out about its current viewable centre point by a factor of 2 for each click of the mouse.
- **1:1** - Restores the canvas to its original size

14.6 Grid Tools

The Grid Toolbar enables you to create objects and align them to a grid. Go to '**View**' on the top toolbar and 'check' the '**Grid Tools**'.

If you click on the '**Show Grid**' icon  on the Grid Toolbar then a Grid will appear where each square is 10 pixels.

If you click on the '**Snap to Grid**' icon  on the Grid Toolbar then any Object that is now added to the canvas will automatically align itself to the nearest gridline in both the horizontal and vertical positions.

You can define the Grid spacing by either selecting '**Options**' found under '**View**' on the top toolbar and clicking on the '**Grid**' tab or clicking on the '**Show Grid Options Dialog**' Icon .

In the presented panel you can define the Grid display spacing. Note:- If you reduce the grid to 1 Pixel then you may not be able to see the grid unless you zoom in.

14.7 Alignment Tools

The Align Toolbar enables you to align objects on the canvas in various formats. To activate this function go to '**View**' on the top toolbar and 'check' the '**Align Tools**'.

The first task is to select all the objects to be aligned. This can be done by either:-

- Holding down the '**Shift**' key and selecting the objects, or
- Selecting the '**Pick Tool**' icon on the '**Draw Toolbar**' then - on the canvas - drag the mouse to create a rectangle enclosing the objects.

The selected objects can now be aligned as follows



Align selected objects to the left



Align the selected object centrally about the horizontal




Align selected objects to the right



Align selected objects to the top




Align the selected object centrally about the vertical

 Align selected objects to the bottom


 Align selected objects about their centres

14.8 File Tools

The 'File Tools' are standard windows tools except for the following

Duplicate selected Object. To see what this does select an object on the canvas and click on the 'Duplicate selected Objects' Icon  found on the 'File' Toolbar. A duplicate Object will be created. The duplicated Object can now be manipulated as required

If you are going to use this facility then you may wish to define the amount of offset for the duplicated object. To do this select 'view' on the top toolbar and then 'Options'. Under the 'duplicates' tab you can define the horizontal and vertical shifts.


Clone selected Object. To see what this does select an object on the canvas and click on the 'Clone selected Objects' Icon  found on the 'File' Toolbar. This will create a duplicate of the Object but, in this case, any changes to the original Object will also be applied to the Clone

14.9 Translation Tools

The Translation Toolbar enables you to select an object and rotate or mirror the image. To activate this function go to '**View**' on the top toolbar and 'check' the '**Translation Tools**'.

If you now select an Object you can translate it in the following manner:-

 Rotate the selected image 90° in a clockwise direction

 Rotate the selected image 90° in an anti-clockwise direction

 Mirror the selected image along its horizontal axis

 Mirror the selected image along its vertical axis

Create a Triangle with unequal sides. If you select this object and click on each of the above icons in turn you will see the effects

14.10 Inspector

Whenever an Object is created an Inspector record is also created which defines a variety of ways in which the Object should appear and how it should perform. You have already addressed some of the features in the Inspector.

14.10.1 Recap

Let's recap some of these features:-

Brush – Allows you to colour and/or shade the object

Height – lets you define the height of the displayed object

Left – This defines the position of the object from the left-hand edge of the canvas

Pen – Lets you specify the colour, thickness of the border surrounding the object

Roundness – Allows you to 'round' the corners of an object

Top - This defines the position of the object from the top edge of the canvas

Width - lets you define the width of the displayed object

Text – Applies to a Text object and facilitates entry of any Text

Font - Applies to a Text object and facilitates format of your entered Text

Alignment - Adjusts the horizontal position of the Text within the Object

Layout - Adjusts the vertical position of the Text within the Object

Angle - This defines the angle (0° to 360°) at which the Text will be displayed

Wordwrap - If this is set to 'True' then the text will be contained within the Text rectangle.

Transparency - Defines the 'opaqueness' of the object. Note: If an Object has a Transparency Value then that value will determine the Opaqueness. However you will find that, in the Workflow, if you move the cursor over the object it will revert to 100% for the duration that the cursor covers it.

Autosize - Resize a Text box to suit the current Text or to fit an Image

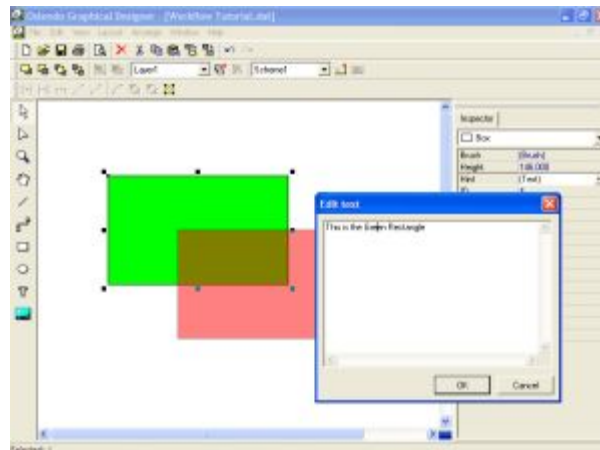
Picture – Applies to a Picture object and links to the image

14.10.2 Hint

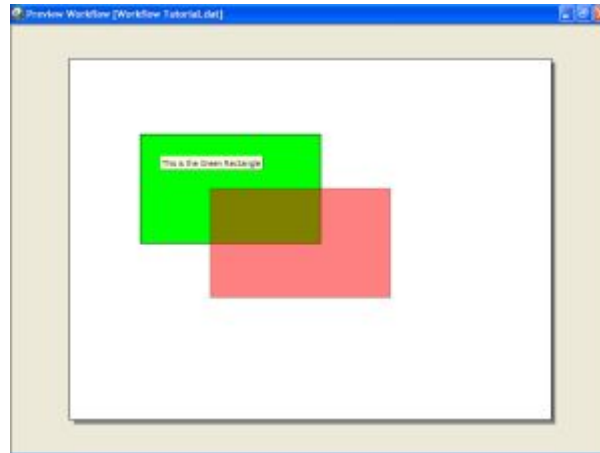
Up to now you have only been in Edit Mode. You can see how the canvas would appear to an end user if you select **Preview** found under **File** on the top toolbar or alternatively click on the

Preview Icon  on the **File** Toolbar. This feature will be used now to demonstrate the Hint.

In Edit Mode click on the Object then the **Hint** drop-down on the **Inspector** Toolbar. A panel will appear for you to enter an open format text. This Text will appear as a **Tool Tip** on the Preview panel.



Go to the Preview panel and place the cursor in the Object. The Hint should now be displayed.



14.10.3 Name

Whenever an Object is created it is automatically given a Name (Curve1, Box1, etc). You can make this more meaningful by selecting '**Name**' on the Inspector Toolbar and entering your name for this object. This name now appears on the Inspector's main drop-down list.

14.11 Layers and Schemes

Firstly let us define a Scheme and a Layer

A **Scheme** is, effectively, another canvas within the same Workflow. You can link schemes together so that they can be viewed relative to each other.

A **Layer** is a restricted view of the content across all Schemes. For example you may have created a single Workflow that spans many Departments. Each Department's activities can be covered within a Layer. Although the combined view shows all Departments you can selectively turn off Layers and make it specific to a Department.

14.11.1 Layers




Begin by activating the '**Layer Manager**' found under '**View**' on the top toolbar. A separate panel will appear on the right beside the Inspector panel.

1. Creating a new Layer

Create a Text Object on the opening Layer and enter Text '**Take a Sales Order**'.

On the '**Layout**' Toolbar you will see a '**New Layer**' icon . Click on this to create a new Layer. You will see that nothing appears to have happened!! You have created a new Layer but all objects on all Layers are currently displayed.

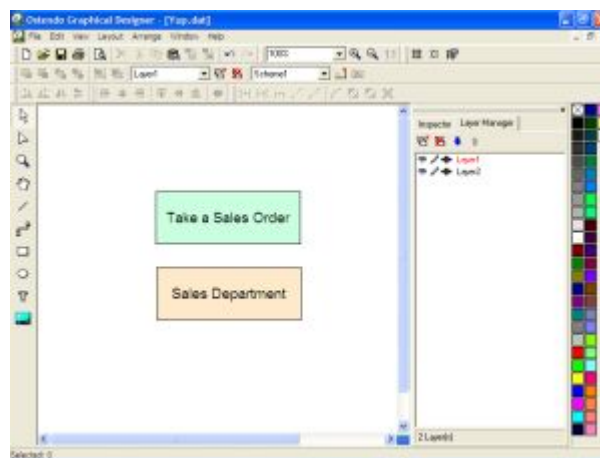
If you go into the Layer Manager down the right hand side you can see the current layers each preceded by three icons. These are:-

-  **Visible** - If this is 'True' then this Layer is visible when viewing the canvas.
-  **Read Only** - If this is 'True' then the objects in the layer cannot be amended.
-  **Moveable** - If this is 'True' then the objects in the layer cannot be moved.

Make the original Layer Invisible (I.e. Click on the '**Visible**' icon) and you should now see a blank canvas in which to work with this Layer. On this Layer add a Text Box and give it text (say) '**Sales Department**' (Note: Field 'Layer' in the Inspector panel should refer to this Layer)

You should note that you do not need to hide other layers to work on the current layer. You simply click on the Layer Name to make that Layer active whilst still viewing all layers.


Make the first Layer Visible and align the two Text Boxes as shown in the screen below



You should note that the Object in the layer is still active even though you cannot see it

14.11.2 Schemes

1. Creating a Scheme

Using the Workflow you used for Layers create a new Scheme by going to the '**Layout**' Toolbar and selecting the '**New Scheme**' icon . A new Scheme will be created. On this scheme add a **Text Object** that details the activities when taking a Sales Order. For example

Select Customer and check their Credit Status. If Credit Limit has been reached report to Supervisor.

Linking Schemes

Go back to the first Scheme (I.e. Select **Scheme1** from the '**Scheme**' drop-down on the Layout Toolbar) and click on the '**Take a Sales Order**' object. In the '**Inspector**' record for this Object click on the drop-down field '**Reference**'. You will see **Scheme1** and **Scheme2**; select **Scheme2**.

If you now select '**Preview**' found under '**File**' on the top toolbar you will see **Scheme1** with the

your **'Take a Sales Order'** object. If you move the cursor to this object it will change to a 'Hand' shape. Clicking on this will send you to **Scheme2**.

Of course you cannot return to Scheme1 because you haven't created a linkage. Therefore go into Scheme2 and click on the button then add the **'Reference'** link to **Scheme1**

There is no limit to the number of links you can have on any scheme

3. Opening Scheme


If you click on the drop-down button located in the **'Schemes'** field you should select **Scheme2**. (Note: You may need to select this twice). Click on the **Inspector** tab on the right of the screen, which will enable you to see the Inspector fields that apply to this Scheme.

If you change field **'Default'** to **'True'** then this scheme will be the opening scheme when you run the Workflow

14.12 Library

Libraries allow you to create Objects or Object Groups and store them in a Library. This Library can be referenced and selected objects dragged and dropped onto the current canvas. To activate this function go to **'View'** on the top toolbar and 'check' the **'Library'** option. A tab will appear beside the **'Inspector'** tab. Select the **Library** tab.

14.12.1 Creating a new Library

On the toolbar in the Library panel click on the 'New Library' icon . This will display a panel for identifying the Library. It is suggested that you maintain separate Libraries for different style Objects such as:-


- Buttons
- Arrows
- etc

Create a Library called (say) **'Library1'** and 'Save' it

Within each Library you can store the relevant Objects. Therefore the next step will be to add objects to the Library

14.12.2 Adding Objects to a Library

Using your current canvas (or create a new one) add an object. This Object will become a 'Standard' Object that we will store in the Library.

Ensure that the Object is highlighted then click on the **'Add selected objects to library'** icon  in the **'Library'** toolbar. Complete the fields on the displayed panel and the object now exists in the Library. Note:- If you do NOT click on the **'Save the Library'** button then it will be lost after you exit Ostendo Graphical Designer, therefore make sure that you carry out this step.

You can also save groups of unrelated Objects. Create a circle and put another circle beside it. Now select both objects (Shift - Left mouse on each). Now save the selection(s) in the current Library.

14.12.3 Adding Library Items to the canvas

If a Library is not yet open you should click on the '[Open Existing Library](#)' icon on the '[Library](#)' Toolbar.

Having selected the Library the objects within it are displayed. If you now select the required Object you will see that it is displayed at the bottom of the 'Library' panel.

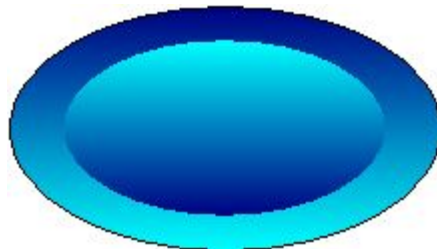
To use this object simply drag and drop it onto the canvas.

14.13 Simple Examples

Try the following exercises to create sample Objects

14.13.1 Oval Button

We will create the following button



To create this carry out the following steps:-

- Create an Oval Object
- On the Inspector panel select 'Brush' and set the following
- Gradient radio button with
 - Vertical
 - Navy
 - Aqua
- Duplicate the Object (Duplicate icon on the File Toolbar)
- Mirror the duplicated Object (Mirror Vertically on the Translate Toolbar)
- Reduce the size of the second Oval Object
- On the Inspector panel select 'Pen' and set the Pen Style to 'Clear'
- Select both objects and click on the 'Aligns the Centres of the selected objects' button'
- Group the two images
- Add the grouped Object to the Library

When you copy the Image to a canvas you can, of course, size it as required.

14.13.2 Corner Pipe

Here is an example of creating a corner pipe

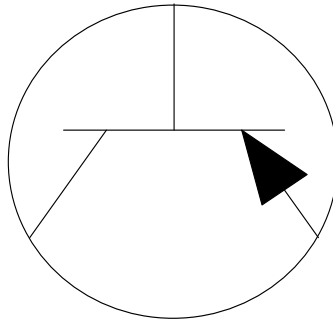


To create this, carry out the following steps:-

- Create an Oval Object with the following settings
 - Begin Angle 90
 - End Angle 180
 - Brush - 'Gradient' Radio Button - Bottom Right with White and Grey
 - Height 150
 - Width 150
 - Pen - Clear
- Create a second Oval Object with
 - Begin Angle 90
 - End Angle 180
 - Brush - 'Standard' Radio Button - Solid White
 - Height 50
 - Width 50
 - Pen - Clear
- Select both objects and
 - Align Right
 - Align Bottom
 - Group them together
- Create a Rectangle Object with
 - Height = 130
 - Width - 10
 - Brush = 'Standard' - Solid Black
- Position the Object against the Pipe
- Click the duplicate button on the Rectangle Object and rotate 90 degrees
- Position the Object against the Pipe
- Group the final picture

14.13.3 Electric Symbol

Here is an example of an electric symbol



- Create an Oval Object with:
 - Height 150
 - Width 150
- On the Inspector panel select 'Brush' and set the following
- 'Standard' radio button with
 - Brush Style 'Solid'
 - Brush Colour 'None'
- On the Inspector panel select 'Pen' and
 - Set the Pen Style to 'Solid'
 - Set the colour to 'Black'
- Create the four straight lines and position within the circle then Group
- Create a straight line which would form the 'base' of the arrow then
 - Click the 'Curve Edit' tool
 - Press Ctrl key and move cursor to form an arrow. Click mouse
 - Shift-Click the two ends and select the 'Close Figure' icon on the 'Curve Edit Tools' toolbar
 - On the Inspector panel select 'Brush' and set the following
 - Standard radio button with
 - Brush Style 'Solid'
 - Brush Colour 'Black'
- Adjust to suit
- Group the objects

14.14 Creating a Process Flow

In this exercise we will create a Process Flow and integrate it with Ostendo. The Process Flow will cover:

- Creation of Items
- Creation of a Bill of Material
- Raising an Assembly Order
- Issuing Components
- Receiving finished Goods

14.14.1 Creating the Process

Start up Ostendo Graphical Designer and click on **File>New** to create a new workflow Graphic. Click on **File>Properties** and amend the **Document Width** to **1040** and the **Document Height** to **600**. (This is the default size for Ostendo Workflows).

We will create the following Process Flow



On the 'Zoom' Toolbar amend the scale to suit your needs.

Create a Text Object on the palette and, in the Object Inspector set:

Alignment = Center
Brush = Select your own colour
Font = Select a suitable Font and font size
Hint = Enter a 'Hint' to appear in view mode
Layout = Center
Pen = Select 'Solid' pen style
Roundness = Enter 10
Text = Enter **Items**

Duplicate the Object and amend Text to read **BOMs**. Give it a different Colour

Repeat for another 3 Objects with the following Text

Object 3 - **Create Order**
 Object 4 - **Issue Components**
 Object 5 - **Receive into Stock**

Draw a connecting line between the first and second object and, in the 'Inspector' panel, select an 'End Cap' Style. Repeat this to connect each Object

Save the Graphic as '**Assembly Workflow**'

Close the Graphic Designer

14.14.2 Display on Ostendo Desktop

Go into Ostendo and click on **File>System Configuration>User Security and Options**. Highlight '**ADMIN**' and select the '**Workflow**' tab. 'Check' the '**Enable Workflow**' checkbox and then click in the middle of the lower panel to activate the '**Add**' button. Click on the '**Add**' button to create a new line. Enter a short 'Caption' to identify the Workflow. In the drop-down under field **Filename** locate '**Assembly Workflow**' and '**Save**' the setting. If you now exit the User Security and Options screen you will find that the Workflow is presented on the Desktop. Note: If the Desktop Views option is also selected then they will appear under their own desktop 'tab'

If you move the cursor over the Object you will see the Hint.

Go to the first Object (**Items**) and 'right mouse'. Click on '**Assign Function**' and then select **Inventory>Items**. You have now linked the Object to the Item Master function. Click on the Object and see what happens.

Repeat for the remaining objects with their respective maintenance screens. I.e.

Object 2 - **Assembly>BOM**
 Object 3 - **Assembly>Assembly Orders**
 Object 4 - **Assembly>Assembly Issues**
 Object 5 - **Assembly>Assembly Receipts**

14.14.3 Using the Process Flow

You can now progress through the screens in a process flow and carry out a practical example

- Object 1 - Add Items **A**, **B**, and **C**
- Object 2 - Create a **BOM** for **A** comprising of **B**, and **C**
- Object 3 - Create an Assembly Order for **A**
- Object 4 - Issue Components **B** and **C**
- Object 5 - Receive parent product **A**

14.14.4 Adding a KPI

This exercise shows you how to add a Key Performance Indicator to the Process Flow.

In the displayed workflow right mouse on the palette (not an object) and select **'Edit Workflow'**. This will bring up the Graphics Designer. What we are going to do is add a Text field that will be constantly updated with a Script. This will show the number of current Assembly Orders with status **'Open'**

Create a Text Object and place it below Object 3 (Assembly Order). Note the Object ID found in the **ID** field of the Object Inspector

'Save' the workflow and exit. Right mouse and select **'Refresh'** to bring the amended Workflow into memory.

Click on **File>Custom Menu Scripts** and add a new script called **'AssemblyWorkflow'**. In the script tab enter the following (or you can simply copy and paste the script shown below)

```
var
  TheResult,TheCaption,TheCount : String;
begin
  TheResult := GetSQLResult('select count(sysuniqueid) from AssemblyHeader where
Orderstatus = "Open" or Orderstatus = "InProgress");
  if (vartostr(TheResult) = "") then TheCount := '0' else TheCount := vartostr(TheResult);
  TheCaption := 'Open Assembly Orders: ' + TheCount;
  setworkflowobjecttext(4,TheCaption);
end.
```

Note: In the above script the line **'setworkflowobjecttext(4,TheCaption);'** the number refers to the Object ID. Change the number to that used by the KPI Object

In the workflow, right mouse on the object and select **'Assign Function'**. Go to, and select, **Custom>Assembly Workflow**. You have now linked the Object to the Custom Script.

Click on the Object and see what happens.

If you right mouse on the Object you can see that you can also define a **'Refresh Interval'** by which Ostendo will re-run the Custom Script

14.14.5 Link an Object to your website

This exercise shows you how to add an object to your Workflow which, when clicked, will open up your website.

In the displayed workflow right mouse on the palette (not an object) and select **'Edit Workflow'**. This will bring up the Graphics Designer.

Create a Text Object and add text that is your Web Link Example: www.development-x.com)

'Save' the workflow and exit. In the Ostendo Workflow right mouse and select 'Refresh' to bring the amended Workflow into memory.

Click on **File>Custom Menu Scripts** and add a new script called 'WebLink'. In the script tab enter the following (or you can simply copy and paste the script shown below)

```
begin
  Run('http://www.development-x.com/');
end.
```

Of course amend the web link address to suit your own Web Address.

In the workflow, right mouse on the object and select 'Assign Function'. Go to, and select, **Custom Scripts>WebLink**. You have now linked the Object to the Custom Script.

Click on the Object and see what happens.

If you, once again, right mouse on the object you will see an option called 'Store Script' If you select this then the complete script will be stored in the 'User Date' field of the Object's Inspector within the Graphical and View Developer. This is where it will reside until you go into this field and manually delete the linked script. As it is now embedded in the Object you can go back into Ostendo and under **File>Custom Scripts** delete the Script **WebLink**.

If you click on the Object in the Workflow you will see that it still actions the embedded script even though the script does not now exist in Ostendo.

15 14. Report and View Developer

The Report and View Developer allows you to create and/or amend Reports, Analysis Views, Charts, and Pivot Views within Ostendo and include these in the Ostendo Menu structure. If you go into the Report and View Developer via **File>Reporting Configuration>Report and View Developer** you will see that all the records are held at 'System' Level. You cannot delete or amend these.

If you wish to amend any Report then you must first copy it to the 'Company' folder and amend it from there. Similarly any new Report is created directly in the 'Company' folder. When Ostendo is printing a Report it first checks your 'Company' folder to see if it exists and, if so, it will use that report. If it doesn't exist in your 'Company' folder then Ostendo will use the 'System' level report.

15.1 Reports

Before we begin you should note the following:

If you go into the Report and View Developer via **File>Reporting Configuration>Report and View Developer** you will see all the Reports, Views, Charts and Pivot Views supplied with Ostendo are held at the 'System' Level. Any new Report, Analysis, Pivot, Graphical, or Inquiry View - whether you create it 'From Scratch' or 'By Copy' - is automatically created with a Level of 'Company' and will apply specifically to the current Company you are signed into. .

Additionally, if it is a Report, then the Report Layout will be added as an .fr3 file to an Ostendo 'Reports' sub folder where the folder name equates to your current Company

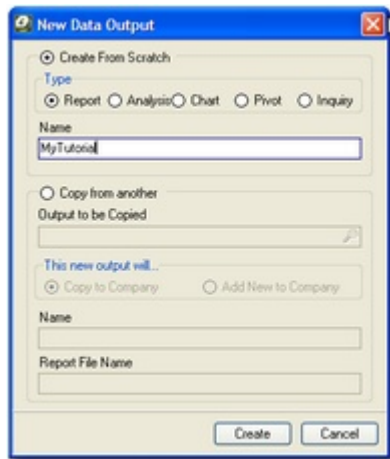
15.1.1 The Report

In these Exercises we will begin by creating a simple Report and progressively include more functionality to demonstrate both the simplicity and complexity of Ostendo's Report Writer.

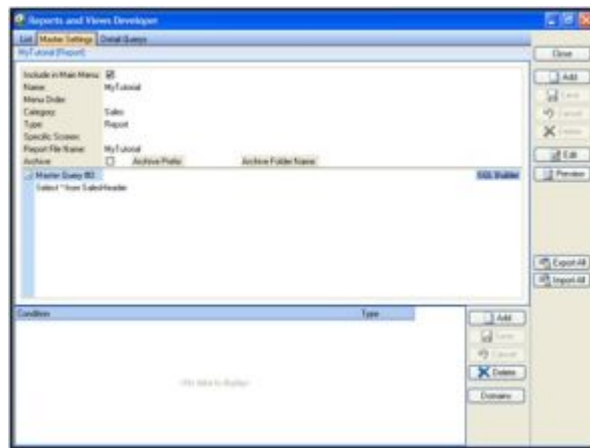
15.1.1.1 Defining the Report

We'll start off this Tutorial by creating a simple 'Sales Order Listing' report using basic functions within the Report Writer.

Go into **File>Reporting Configuration>Report and View Developer** and click the '**Add**' button to create a new Report. 'Check' the '**Report**' Radio Button and enter a Report name of (say) '**MyTutorial**'.



Click the '**Create**' Button to take you to the Master Settings tab to define the Reporting requirements.



In this screen enter the following:

Include in Main Menu: 'Check' this checkbox. This will place the Report under the Reports menu of the Main Toolbar selection identified under 'Category'.

Name: Prefilled with the name you entered into the 'New Data Output' panel (above). You may alter this if you wish.

Menu Order: This defines the position in the Drop-Down menu of all Other reports in the same Category. For now leave this blank.

Category: This defines under which Main Toolbar Menu Item this report will sit under. Therefore select '**Sales**' from the drop-down list.

Type: This can be 'Report' or 'Form' (see Section 8 for the difference between these). For now select **Report**

Specific Screen: You can also make this Report available under the 'Reports' button in a selected screen. We will make this available under the Sales Order screen. Therefore select Sales Orders from the drop-down list.

Report File Name: This is the name (followed by .fr3) that holds the Report Layout details. This will be held in a folder identified as you current signed-on company which is located under Ostendo's main Report Folder

Archive: This is only used if you wish to take an archive copy each time this report is run. This is usually associated with 'Formal' documents such as Invoices, Statements, etc.

Master Query: In the Master Query area type in the following query **Select * from**

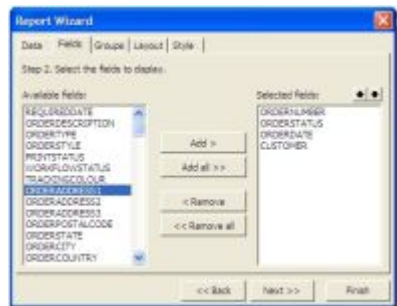
SalesHeader. This will select all fields that are currently held in a Sales Order Header record. You can click on the SQL Builder button to the top-right of this field to help you create and maintain queries.

Click the '**Save**' button to save the entry

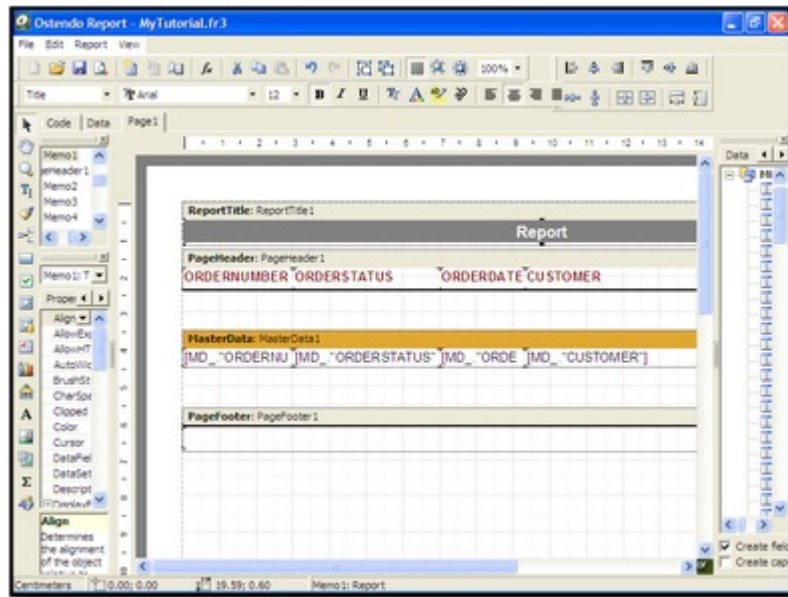
You have now defined the Report Name, where it is to appear and what data is being sent to the Report itself. What is now required is to create the layout of the Report. Therefore click on the '**Edit**' button. As this Report does not currently exist a small panel will be presented that gives you the option to use a 'Wizard', copy an existing layout, or commence with a blank palette. We will use the Wizard. In the '**Items**' tab select the '**Standard Report Wizard**' icon and click the **OK** Button.



In the presented panel click on the '**Fields**' tab and transfer **ORDERNUMBER**, **ORDERSTATUS**, **ORDERDATE**, and **CUSTOMER** to the right-hand panel. Before you click the '**Finish**' button have a look at the content of the remaining tabs



Once you click the '**Finish**' button the Wizard will generate the Report based on these selections.



If you now click on the 'Preview' Icon (4th from left on the top toolbar) you will see the finished printed report. Exit out of Report and View Developer (saving where required) and then go to **Sales>Reports** you will see that your 'MyTutorial' report is listed.

15.1.1.2 Amending the Report

Having created the basic Sales Order Listing we will look at some of the changes that you can do to amend the Look and Feel of the report

First of all you can see that all data is contained within 'Bands'. In the above report we have 4 Bands


- **Report Title:** This is printed once at the beginning of the Report
- **Page Header:** This is printed at the beginning of each Page in the Report
- **Master Data:** This is printed once for each record selected in the Query
- **Page Footer:** This is printed at the bottom of each Page in the Report

Let's look in more detail at each Band to see what we can do with them. You should note that this is only an introduction to each band and does not cover all available options.

1. Report Title Band

The displayed Band has two portions. The upper portion tells you that it is a 'ReportTitle' Band and the lower portion is where you place the Title itself. If you click on the word ReportTitle then six 'Handles' will appear covering the lower part of the band. You can drag the centre handles up/down to increase or decrease the space occupied by this band in the finished report.

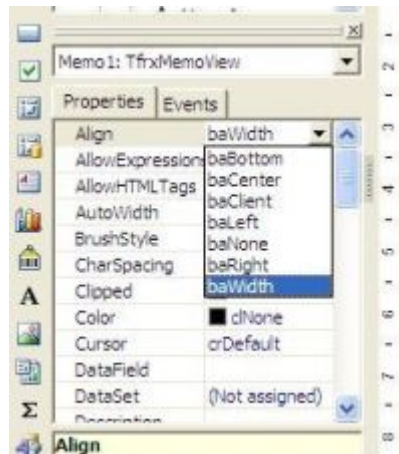
'Right Mouse' on the word 'Report' in this band and select 'Delete'. What you have just done is selected a 'Text' object that was in the Band and removed it. What we will do now is recreate it.

Click on the 'Text' object  found down the left of the screen. Move the mouse into the Title Band and drop the object. A panel will be presented into which you can type the words '**Sales Orders Report**'. Click the **OK** button to exit. Now let's format the object.

Click on the **'Align'** object on the **Text** Toolbar to align the Text centrally in the object



Down the left of the screen you will see the **'Properties'** tab that displays the properties which apply to this Object.



Under property **'Align'** select **baWidth**. Next go to **'vAlign'** and select **'vaCenter'** to vertically align the Text within the object. Now grab the 'handles' on the object and fit to suit the depth of the **ReportTitle** Band

You will also see, in the above **Text** Toolbar, that you can format the font style and size of the text. Amend this to suit your requirements

Finally go back to the **'Properties'** panel and select a background colour from the drop-down list against **'Color'**.

2. Page Header Band

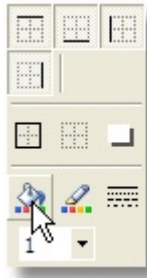
The Wizard pre-populated this band with 'Text' objects **A** each reflecting the selected field name. You should double-click on each object and amend the Text to be more meaningful. For example: **'ORDERNUMBER'** should be amended to **'Order Number'**.

If you scroll to the right and click on the space to the right of the **'Customer'** object you will see that all these Text Objects have been superimposed on top of a further 'Text' object that simply displays a single – broad – line. Select this object.



Now go to **View** on the top toolbar and select **Toolbars**. From the drop-down list select **'Frame'**.

You can now dictate how the 'frame' of this Text object is displayed. Try some of the available options



3. Master Data Band

The Wizard populates this band with the selected fields. As the field is enclosed in 'square' brackets the Report Writer knows that this is a database field and the MD_ tells it which Query it relates to (In this instance the Master Data Query). The field definition is replaced with data for each record selected. For example: **[MD_."CUSTOMER"]** will be replaced by the Customer Name of the record being printed.

The attributes that you looked at for a 'Text' field in exercise 2. also apply to Data Fields

4. Page Footer Band

If you scroll down to the Page Footer Band you will see that it can contain information about the report such as Page Number, Date, Time. If you double-click on the 'Page' Object you will see that it comprises of a combination of Text and Database fields. The Report Writer will know that the first word 'Page' is text because it is not enclosed in 'Square' Brackets whereas **[Page#]** is.

The **[Page#]** is a pre-defined Report variable and, if you look down the right-hand side of the screen you will see a tab called 'Variables' in which you can see other variables. Drag and Drop one of the Variables into this Band then 'preview' the Report to see what happens.

If you know the Variable ID's then you can string together a combination of Text and Variables. For example: Amend the Page Object from

Page [Page#]
to
Page [Page#] of [TotalPages#]

4. Adding more Data Fields

Up to now you have let the Wizard decide which fields you wish to see in the Report. In this exercise we will add more data to each printed record.

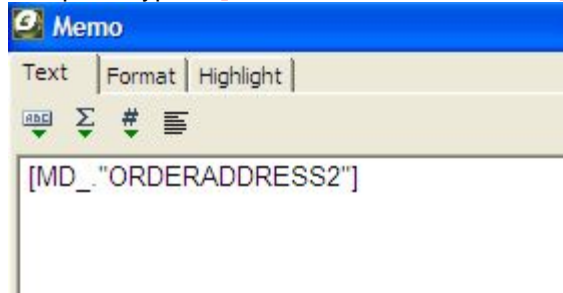
If you look down the right-hand side of the screen you will see a tab called 'Data' in which you can see all the fields extracted by your Master Data Query **Select * from SalesHeader**. We will add two or three of these to the Report.

Go to the bottom edge of the **MasterData** Band and drag it downwards to increase the depth of the band.

From the 'Data' tab drag and drop **ORDERADDRESS_1** field immediately under the Customer field as shown on the screen shot below. At this stage do not add Address Lines 2 or 3. There are two more methods by which you can add Data Lines to the Report. We will add Address Lines 2 and 3 using these:


Method 1

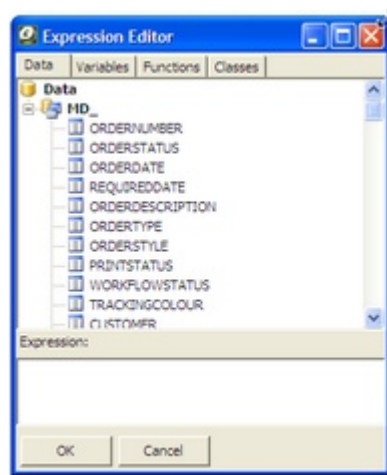
- Add a 'Text' Object and place it under ORDERADDRESS_1
- In the presented panel type in **[MD_"ORDERADDRESS2"]** and click the **OK** Button



- Size the Object to the required width by dragging the edges of the Object

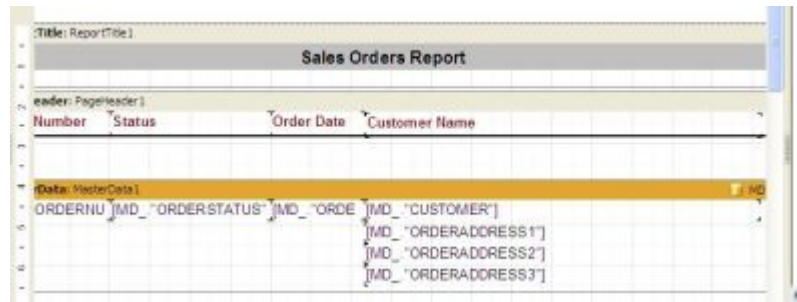
Method 2

- Add a 'Text' Object and place it under **ORDERADDRESS_2**
- In the presented panel click the 'Insert Expression' Icon  at the top-left to bring up the **Expression Editor** panel



- Locate **ORDERADDRESS3** and double-click on it to send it to the lower 'expression' area.
- click the **OK** Button
- click the **OK** Button once more

Your final layout should look something like this



If you 'Preview' the report you will see that the Address for each Sales Order record will be printed.

5. Page and Output Settings

While you are in the Report Layout screen let's have a look at the Page Sizing and Printer options. Click on

- (a) **File>Page Options**; This allows you to define the size, orientation and margins to be used in the Report
- (b) **Report>Options**: To define the preferred output Printer. You will need this if you are (for example) printing to a Label Printer

Save the settings and close out of the report 'Edit' area to go back to the **'Master Settings'** screen

15.1.1.3 Adding Detail Lines

You may wish to add Sales Order Lines to the Report and get the lines for each Sales Order will print out immediately after the associated Sales Order Header.

1. Defining the Query

The first task is to create a Query that pulls in all Sales Order Lines. To do this you should go back to the **MasterData** screen and click on the **'Detail Query'** tab. In that screen click on the '+' beside **'Query #1'**. This will open up a 'Query Area' into which you can enter your query to extract the Sales Order Lines.



Enter the following Query

Select * from SALES LINES where ORDERNUMBER = :ORDERNUMBER

Let's look at the Query. It is a standard query except for the final **:ORDERNUMBER**. This is telling the query to only extract the Sales Order Lines where the **ORDERNUMBER** field in the Order Lines Table equals the **ORDERNUMBER** field in its 'parent' record. The 'Parent' record is defined in the lower left of this 'Query Area'. In this instance it is linked to Query #0 (i.e. The Master Query) and therefore refers to the **ORDERNUMBER** field in the Sales Order Header.

Now go back to the 'Master Settings' tab and click on the 'Edit' button

2. Adding the Sales Order lines


There are Four steps that now need to be carried out

- Make the Sales Order Line fields available
- Add the Sales Order Line fields
- Add a Header Band to define the Order Lines
- Add a Footer Band to denote the end of the Order Lines

1. To make the Sales Order Lines available click on **Report>Data** on the top toolbar and 'check' the checkbox against **DD_1** (I.e. Detail Data Query Number 1). If you now look at the 'Data' tab down the right-hand side of the screen you will see that it now contains all the Sales Order Line fields under **DD_1**


2. Adding the Sales Order Line fields



- Down the left of the screen click on the **Insert Band** Icon 
- Select '**Detail Data**' from the drop-down List
- On the presented panel link it to **DD_1**
- From the '**Data**' tab down the right-hand side drag and drop **CodeType**, **LineCode**, **OrderQty**, **OrderUnitPrice**, and **ExtendedTotalPrice**.
- Position the fields as required. You may wish to Right-Align the last three fields by clicking in the field and selecting Right-Align on the '**Text**' Toolbar
- For the two Currency fields you should double-click on each field and select the '**Format**' tab in the presented panel. Select '**Number**' under '**Category**' panel and **\$1234.50** under '**Format**' panel. The field will now be displayed as a Currency Field using your home currency symbol

3. Adding a Header Band

You should now add a Header Band and populate with Text Objects that describe the fields in the DetailData Band

- Down the left of the screen click on the **Insert Band** Icon 
- Select '**Header**' from the drop-down List
- Position the Band between the **MasterData** and **DetailData** Bands
- Add a '**Text**' Object into the band and enter Text '**Code Type**'. Position it above the **Code Type** data object in the Detail Band.
- Repeat for the remaining Detail Band Objects
- We will now create a Line across the Band. Therefore add a further **Text** object into the Band. While the object is still selected go to the '**Properties**' tab down the Left and select


'baNone' against 'Align'. Finally, in the 'Frame' Toolbar select a size 2 line covering the bottom of the Text Object
Alternatively you could click on the 'Draw' icon at the bottom of the left-hand toolbar and select 'Line Object'.

The end result should look something like this

Click the 'Preview' Icon on the top toolbar and view the results. Adjust the Report Layout as required.

4. Adding a Footer Band

We will now create a Line across the Report after the Detail Lines have been printed

- Down the left of the screen click on the **Insert Band** Icon 
- Select 'Footer' from the drop-down List
- Place the Band just below the **DetailData** Band
- Add a 'Text' Object into the band. While the Text object is still selected go to the 'Properties' tab down the Left and select 'baWidth' against 'Align'. Finally, in the 'Frame' Toolbar select a size 2 line covering the bottom of the Text Object. Maybe you wish to define the style of the line while in this panel

The end result should look something like this

Click the '**Preview**' Icon on the top toolbar and view the results. Adjust the Report Layout as required.

15.1.1.4 Optional Printing

A Sales Order Line contains a '**Notes**' field that can hold up to 65,000 characters of information. Obviously you do not want to allocate space for this in the report if the field doesn't contain any Notes. Also, if it does contain Notes then the printout should vary in size to accommodate the amount of Notes. To demonstrate this we are going:

- To create a Child Band linked to the Sales Order Line
- Add the 'Notes' field to that Band
- Make that Band display only if there are Line Notes
- Vary the size of the printed field to conform to the Notes content

In preparation for this go into a Sales Order Line and add lots of Notes that would print out on multiple print lines. For example

These are Notes added during the Training Session
This is Line 2 Notes
This is Line 3 Notes
This is Line 4 Notes
No more Line Notes

Go back into the **Report and View Developer** and select your report. Go to the **Master Settings** screen and click the '**Edit**' button. Enter the above Sales Order in both the **From** and **To** parameters and go to the Report Layout

Click on the Band Icon and select '**Child**'. Position the Band underneath Band **DetailData1**. To link the two bands together select the **DetailData1** Band and go to its '**Properties**' down the left of the screen and select '**Child1**' from the drop-down under '**Child**'.

Go to the '**Data**' tab down the left of the screen and locate **LINENOTES** under **DD_1**. Drag and drop this into the Child Band and '**Preview**' the Report. You will notice that it only displays the content that can fit into the allowable space defined by the Object. Therefore drag the left and right-hand edges of the Object and size to suit the printout.

The Report Layout should look something like this

ReportTitle: ReportTitle				
Sales Order Report				
PageHeader: PageHeader1				
Order Number	Status	Order Date	Customer Name	
MasterData: MasterData1				
[MD_ "ORDERNU"]	[MD_ "ORDERSTATUS"]	[MD_ "ORDE	[MD_ "CUSTOMER"]	
			[MD_ "ORDERADDRESS1"]	
			[MD_ "ORDERADDRESS2"]	
			[MD_ "ORDERADDRESS3"]	
Header: header1				
Code Type	Code	Ord Qty	Unit Price	Extended Price
DetailData: DataData1				
[DD_1 "CODET	[DD_1 "LINECODE"]	[DD_1 "ORD	[DD_1 "OR	[DD_1 "EXT
Child: Child1				
	[DD_1 "LINENOTES"]			

If you click the **Preview** Icon you will see that it only prints the first line of the Notes. Therefore click on the **[DD_1."LINENOTES"]** Object and refer to the **Properties** tab down the left of the screen. In the properties locate **'StretchMode'** and select **smActualHeight**.

If you again click the **Preview** Icon you will see that although it prints all the Lines from the Notes field the **'Child1'** Band has not stretched with it. Therefore select the **Child1** Band and refer to its **Properties** and amend **'Stretched'** from **False** to **True**

If you run the report now you will find that the Notes appear OK within the Child Band.

Go back to the Master Settings tab and re-run the Report but - this time - select a range of Sales Orders. You will find that each Order Line shows the **[DD_1."LINENOTES"]** field even if there are no Notes. What we must now do is tell the Report Writer NOT to print the Child1 Band if there are no Notes.

Select the **Child1** Band and then click on the **'Code'** tab to the right of the screen. Replace the existing **Begin** and **End** statements by copying (Ctrl-C) and pasting (Ctrl-V) the following script

```

procedure Child1PrintOption(Sender: TfrxComponent);
begin
  if ((<DD_1."LINENOTES"> = null) or (trim(<DD_1."LINENOTES">) = '')) then
  begin
    Child1.visible := false;
  end
  else
  begin
    Child1.visible := true;
  end
end;
End;

Begin

End.
```

The above procedure states that if the **DD_1."LINENOTES"** is null or blank then don't print the Child1 Band.

Having defined the Procedure we now need to tell the Report Writer when to run it.

Click on the **Page1** tab (to the right of the **'Code'** tab) and ensure that you have still selected the **Child1** Band. To the right of the **'Properties'** tab you will see an **'Events'** tab. This is where we tell the Report Writer **'WHEN'** to run the Procedure. Click on this tab and go to field

OnBeforePrint. From the drop-down in the adjacent field select your Procedure **Child1PrintOption**

If you **Preview** the Report you will find that it prints out as required. Having added code to the Child1 Band you will now see a small red triangle to the left of the word '**Child**' in the Child Band to denote that it has associated code.

That's it !!! Just to recap. You have defined a Procedure of the action to want to take and then defined when that action should be carried out.

15.1.1.5 Report Totals

We will now create Totals at both the Report and Sales Order levels


1. Report Total

We will now create a Report Total that displays the sum of all the selected Sales Order Lines. The finished layout should look something like this

Code Type	Code	Qty	Unit Price	Extended Price
DetailData: DetailData1				
[DD_1^CODE]	[DD_1^LINECODE]	[DD_1^ORD]	[DD_1^ORD]	[DD_1^EXTEN
Child: Child1				
[DD_1^LNENOTES]				
Footer: Footer1				
ReportSummary: ReportSummary1				
Report Total				[SUM(<DD_1^
PageFooter: PageFooter1				
				Page Page#

To achieve this do the following

Step 1. Insert a Report Summary Band

Down the left of the screen click on the Insert Band Icon . Select '**Report Summary**' from the drop-down List

Step 2. Add a Field Label

Add a '**Text**' Object into the band and enter a Text of '**Report Total**'. Format the field as required – example Font Size, Bold, etc)

Step 3. Add the Report Summary field

Add another '**Text**' Object to the right of the above object and - in the presented panel – carry out the following actions:

Click on the '**Insert Aggregate**' Icon  to bring up the following panel.



Enter the following

- **Function:** Select **SUM**.
- **Data band:** Data comes from **DetailData1** Band.
- **Dataset:** Data Field comes from **DD_1**.
- **DataField:** Select **EXTENDEDTOTALPRICE**.
- Click the **OK** button.
- Click the **OK** button in the next screen to return to the Report Layout.
- Format the field as required.

If you now Preview the Report you will see that it now contains a Report Total of the Extended Prices

2. Sales Order Totals


Although the Sales Order Header (**MD_**) contains the Order Total we will get the Report Writer to calculate this from the sum of the Order Lines.

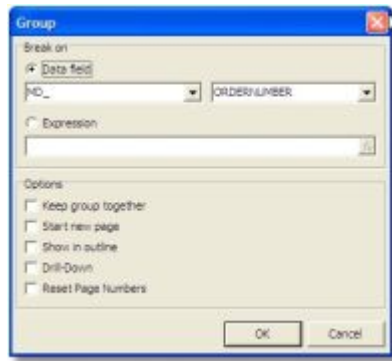
To achieve Order summary totals we need to segregate each Sales Order. This is achieved by adding a 'Order By' statement in the Master Query and – in the Report Layout – separate the Sales Orders by using the Group Band.

Step 1 is to exit the Report Layout and return to the **Master Settings** tab. Amend the main query to read

**Select * from SalesHeader
Order by OrderNumber**

This will ensure that the data being presented to the Report Writer will be in Sales Order Sequence


Step 2. Go back into the Report Layout and click on the Insert Band Icon . Select '**Group Header**' from the drop-down List. The following panel will appear.



into which you should enter the following

- **Break On:** 'Check' the **'Data Field'** Radio Button.
- **First drop-down:** Field comes from **MD_**.
- **Second drop-down:** Field is **ORDERNUMBER**
- Click the **OK** button

You will see the GroupHeader1 Band has been placed in the lower part of the Report. Drag this Band and drop it above the MasterData Band. As we are not placing any data into the Group Header Band you can 'close-up' the space in the band to just leave the Band Title

Step 3. Down the left of the screen click on the Insert Band Icon . Select **'Group Footer'** from the drop-down List. Drag the Band and drop it just below the **'Footer'** Band.

Step 4. Add the two fields. You can do this in a similar manner to how you did it in 7.1. (Report Summary Total) or you can copy existing fields as follows

- **Field Label:** Right click on the field label Object in the **Summary Total** band and select **'Copy'**. Right Mouse and select **'Paste'** then drop the field in this Band. Double click on the Object and amend the Text to read **'Order Total'**
- **Order Summary field:** Right click on the **Report Summary** field Object in the **Summary Total** band and select **'Copy'**. Right Mouse and select **'Paste'** then drop the field in this Band. That's all that is required. Because it is now in a Group Band the Report Writer knows to sum this field for the Group. I.e. Each Sales Order

Step 5. Preview the Report and you will see both Group and Report Totals. At this stage you may wish to adjust the Report to aesthetically suit your requirements

15.1.1.6 Reports -v- Forms

In the Master Settings Tab there is an option to identify the output as a **'Report'** or a **'Form'**.

Let's have a look at the difference between these two

Reports: A 'Report' is what you have been doing above and is all driven from the Main and Detail Queries. The Report will be listed under the appropriate section defined under the Category field and/or the Specific Screen selected in the Master Settings screen. The parameters provided to the Report are those listed in the Master Settings screen

Forms: A 'Form' can roughly be described as an 'External Document' (Example Invoice, Purchase Order, Quotation, etc). These specific Forms are supplied with data from an Ostendo Program and cannot be created by the User.

To see the available forms you should first copy a 'Form' (For example: Purchase Order) to your Company Folder. Therefore go into the **Report and View Developer** and click the 'Add' button. In the presented panel 'check' the 'Copy from another' Radio Button and locate the 'Purchase Order' and click the 'Create' button.

In the 'Master Settings' tab on the generated Company **Purchase Order** you will see that it is a 'Form'. The available 'Forms' can be seen from the drop-down list under 'Name'. You cannot change the name of the Form from (say) **Invoice** to **RevisedInvoice**. This is because Ostendo only provides data to Forms with these Names. You will notice that 'Forms' contain only one condition parameter and this is called SCREENPARAM. It is this value that is passed from Ostendo to define the specific Form Number (Sales Order Number, Purchase order Number, Delivery Number, etc)

Therefore you can only have the one Master Settings for an Invoice, Purchase Order, Job Order, etc. This brings up the question of "But what if I want a different Invoice Layout for selected Customers"

You should note that the Master Settings screen simply defines the data from the Master Query and Detail Queries. Using this extracted data you can create and maintain multiple Forms in the Report Editor area as described in the next paragraph

Going back to the copied **Purchase Order** you can make any changes to the layout as required to suit your requirements. If you wish to have an alternate Purchase Order layout for use against specific Suppliers then you can achieve this as follows.

1. Make the necessary changes then go to **File** on the top toolbar and select **Save As** then save it with another name (example **SpecialPO**). You now have two Purchase Order layouts – the original **PurchaseOrderSheet** and the new **SpecialPO**. You can amend either of these by going to **File** and selecting 'Open'
2. The next task is to link the new Purchase Order Layout to Specific Suppliers. In Ostendo go to **File>Reporting Configuration>Specific Form Layouts**. Click the Add button to create a new record then complete the fields as follows:

Print Form Name: Select Purchase Order
Condition: Select Supplier
Value: Select a single Supplier
File Name: Point it to your new report SpecialPO.fr3

Save the entry.

Whenever a Purchase order is produced for the above Supplier the Report program will use the common Master Settings to extract the data but will use the Specific Form Layout for the nominated Supplier

15.1.1.7 User-Defined Parameters

In addition to having parameters that let you select specific or a range of records you can also create your own parameters against which the user enters data which you then use to control aspects of the printout. Here are a couple of examples of this

15.1.1.7.1 Print Detail

Continuing from the previous exercises we will add an additional parameter to let the User exclude the Order Lines from the print. There are three steps to be carried out:

- Define the Condition parameter
- Create a Report Variable to hold the result of the Condition

- Create Code to manipulate the Report in line with the entry

Step 1. Define the Condition parameter

Go into the Master Settings Tab for the Report and add a new Condition as follows

Print Detail;{Boolean} **9000**

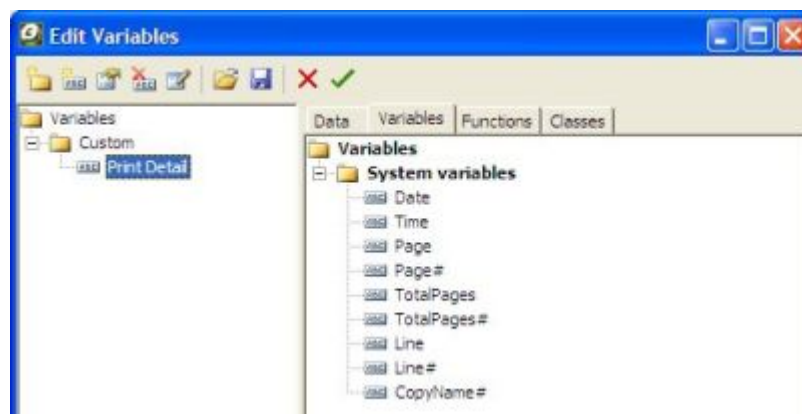
Let's have a look at this in more detail

- **Print Detail:** Open Format text that will be presented when running the Report to describe what is required to be entered
- **;** Separates the above Text from the next field
- **{Boolean}** This is the format of this entry {contained in 'squiggly' brackets}. Data entry is validated against this format although the value itself is stored as a String in the database. The validation options are:
 - Boolean
 - Integer
 - String
 - Date
 - Time
 - Double
- **9000** The 'Type' of Parameter being entered. For User-defined entries this is always 9000

Step 2 Create a Report Variable

Go into the Report Layout and click on '**Report**' on the top toolbar and select '**Variables**' from the drop-down list. In the presented panel click on the '**Category**' Icon (first Icon on the top toolbar). This will create a new Variable called '**New Category 1**'. Single click on this and change the label to **Custom**. Click the Green 'Tick' Icon to confirm the entry.

Having set up the Category select it then click on the '**Variable**' Icon (second from the left on the top toolbar). This will create a new Variable called '**New Variable 1**'. Single click on this and change the label to **Print Detail**. The final screen should look like this



Click the **Green** 'Tick' Icon to confirm the entry and exit the screen

Step 3 Create the Code

We now need to create some code to print the relevant bands depending upon the entry made

against parameter '**Print Detail**'. Click on the '**Code**' tab and replace the current code with this

```

procedure Child1PrintOption(Sender: TfrxComponent);
begin
  if ((<DD_1."LINENOTES"> = null) or (trim(<DD_1."LINENOTES">) = '')) then
  begin
    Child1.visible := false;
  end
  else
  begin
    Child1.visible := true;
  end
  End;
End;

Begin
  if <Print Detail> = 'Yes' then
  begin
    DetailData1.visible := true;
    Header1.visible := true;
    Footer1.visible := true;
    GroupFooter1.visible := true;
    ReportSummary1.visible := true;
  end
  else
  begin
    DetailData1.visible := false;
    Footer1.visible := false;
    Header1.visible := false;
    GroupFooter1.visible := false;
    ReportSummary1.visible := false;
  end;
End.

```

15.1.1.7.2 Number of Printed Copies

In the example we will ask the question as to how many copies of the report you want printed. You should note that we are asking the quantity via a parameter but you could easily use a quantity held against a data record.

Step 1. Define the Condition parameter

Go into the Master Settings Tab for the Report and add a new Condition as follows

Number of Copies;{Integer} **9000**

Step 2 Create a Report Variable

Go into the Report Layout and click on '**Report**' on the top toolbar and select '**Variables**' from the drop-down list. In the presented panel click on the '**Variable**' Icon (second from the left on the top toolbar). This will create a new Variable called '**New Variable 1**'. Single click on this and change the label to **Number of Copies**.

Step 3 Create the Code

We now need to create some code to get the Report Writer to print multiple Copies. For this purpose there is a specific function called **Report.PrintOptions.Copies**. Therefore insert the following code before the final **End.** statement

```
begin
  Report.PrintOptions.Copies := <Number of Copies>;
end;
```

If you run the Report back to the screen you will only see one copy but if you route it to a Printer the Printer 'copies' field will be prefilled with the entered Parameter quantity.

That concludes this Tutorial which was based around Sales orders. The following are other Report Writer features that you may find useful

15.1.1.8 Other Report Features

Enter topic text here.

15.1.1.8.1 Freespace

This determines the amount of Free Space in the current page of the report. You can compare this with (say) a Band to see if that band can fit in this space and, if so, print the band otherwise go to the next page. In Ostendo you will find (for example) that this is used to print out the Order Totals at the bottom of a Purchase Order. Here is an example of how it works using the Report you created above.

What we will do is place the Sales Order Group Footer you generated earlier at the bottom of the Page. As the Order Lines can go over many pages we need to determine – after the lines have been printed – if there is sufficient space to print the Summary Totals. If there is then print it, else go to the next page.

In the 'Code' tab in the Report add the following code to the beginning of the existing code.

```
procedure SummaryTotalOnBeforePrint(Sender: TfrxComponent);
begin
  if engine.FreeSpace <= (GroupFooter1.height + Page1.BottomMargin +
  PageFooter1.Height) then
    begin
      engine.NewPage;
      engine.CurY := engine.PageHeight - (GroupFooter1.height + Page1.BottomMargin +
      PageFooter1.Height);
    end
  else
    begin
      engine.CurY := engine.PageHeight - (GroupFooter1.height + Page1.BottomMargin +
      PageFooter1.Height);
    end;
end;
```

Let's see what is contained in this code:

- The '**engine.freespace**' is a system parameter that holds the current available space per page. We need to determine if this accommodates the sum of GroupFooter1 Band height, the PageFooter Band height, and BottomMargin height . Note: Margins are defined in **File>Page Settings**.
- '**engine.NewPage**' is a system parameter that tells the Printout to go to a New Page
- '**engine.PageHeight**' is a system parameter that holds the height of the page
- '**engine.CurY**' is also a system parameter where you tell the Printer to go to that line within the Page so that you can begin printing the content of **GroupFooter1**

Having entered the Procedure into the Code we now need to tell the Report Writer when to run it.

Click on the **Page1** tab (to the right of the 'Code' tab) and ensure that you have selected the **GroupFooter1** Band. To the right of the 'Properties' tab you will see an 'Events' tab. This is where we tell the Report Writer 'WHEN' to run the Procedure. Click on this tab and go to field **OnBeforePrint**. From the drop-down in the adjacent field select Procedure **SummaryTotal**

If you **Preview** the Report you will find that it prints out with the content of the GroupFooter1 band at the bottom of the page.

15.1.1.8.2 Using the Two-Pass Option

In this exercise we will use the Report's 'Two-Pass' facility to evaluate the ABC category of Items based on their current stock percentage (by value) of the overall Inventory Value.

Firstly let us define what the ABC Category is. It is a form of analysis where Inventory items are divided into three categories (A, B, and C), according to a criteria such as revenue generation, turnover, or value. In this exercise we will use the current Inventory Value.

Typically, 'A' items represent (say) 60 percent of the total value of Inventory. 'B' items represent (say) the next 25 percent (I.e. A and B = 85%), and 'C' Items the balance. This categorisation enables us to carry out a Stocktake where the high value 'A' Items are checked more frequently than 'B' Items which, in turn, are checked more frequently than 'C' Items

We will go through three steps in this exercise

- Extract all Items along with their current stock Value
- Using the Report's Two-pass option the first pass will evaluate the total Inventory Value
- Using the second pass we will categorise each Item into an ABC Category

Also, in the next Exercise, we will provide the option to update Ostendo Item Records with these categories

1. Creating the Master Query

Create a new Report called (say) '**ABC Categories**' and, in the 'Master Settings' screen define the following Master Query

```
Select ItemCode, ItemDescription, ItemUnit, OnHandQty, AverageCost, cast((OnHandQty * AverageCost) as decimal(13,2)) as ExtendedCost from ItemMaster  
order by 6 desc
```

In this query you should note:

- The Extended Cost of each line is calculated from the On Hand Quantity and Average Cost fields in the Item Master
- The records are presented to the Report in descending sequence of the calculated Extended Cost.

In the Report Layout it is recommended that you click on **File>Page Settings** and 'check' the '**Landscape**' Radio Button. Add the selected fields and amend the title. Your finished report should look something like this:

ReportTitle: ReportTitle1					
ABC Category Report					
PageHeader: PageHeader1					
Item Code	Description	Item Unit	On-Hand Qty	Unit Cost	Extended Cost
MasterData: MasterData1					
{MD_ "ITEMCODE"}	{MD_ "ITEMDESCRIPTION"}	{MD_ "ITEM"}	{MD_ "ONHA"}	{MD_ "AVERAG"}	{MD_ "EXTEND"}
PageFooter: PageFooter1					
					Page

2. Two Pass Report

In the Report Layout view click on **Report>Options**. In the presented panel you will see a checkbox **'DoublePass'** If you 'check' this then the report writer will run through the report twice.

2.1. The First Pass

Click on the 'Code' tab and enter the following

```

Var
TotalValue: Double;

procedure TotalInventory(Sender: TfrxComponent);
begin
  if (engine.FinalPass = False) then
    begin
      TotalValue := TotalValue + (<MD_."ONHANDQTY"> * <MD_."AVERAGECOST">);
    end;
  End;
end;

begin

end.

```

- You can see that we have defined a Variable called **TotalValue**. The extended value of each record is read and added to this Variable to show the total value of Inventory.
- You should also note that we have introduced a new system parameter called **engine.FinalPass** against which you specify which pass this action is to be carried out

To show the content of this field you may wish to add a **'Report Summary'** band and populate this band with this variable as shown below

ReportTitle: ReportTitle1					
ABC Category Report					
PageHeader: PageHeader1					
Item Code	Description	Item Unit	On-Hand Qty	Unit Cost	Extended Cost
MasterData: MasterData1					MD
[MD_"ITEMCODE"]	[MD_"ITEMDESCRIPTION"]	[MD_"ITEM"]	[MD_"ONHA"]	[MD_"AVERAG"]	[MD_"EXTEND"]
ReportSummary: ReportSummary1					[TotalValue]
PageFooter: PageFooter1					Page

If you run the Report it will show the total Value of Inventory

2.2. The Second Pass

We will start off by using 'hardcoded' percentages against which to apply the ABC Code and then make them variable by adding run parameters. Replace the current code with the following

```

Var
TotalValue, LinePercentage : Double;
ABCCode : String;

procedure TotalInventory(Sender: TfrxComponent);
begin
  if (engine.FinalPass = False) then
    begin
      TotalValue := TotalValue + (<MD_"ONHANDQTY"> * <MD_"AVERAGECOST">);

    End
    else
      begin
        LinePercentage := LinePercentage + (<MD_"ONHANDQTY"> *
<MD_"AVERAGECOST">);
      End;

    If ((LinePercentage / TotalValue * 100) < 60) then
      begin
        ABCCode := 'A';
      end
    else
      begin
        If (((LinePercentage / TotalValue * 100) >= 60) and
((LinePercentage / TotalValue * 100) < 85)) then
          begin
            ABCCode := 'B';
          end
        else
          begin
            ABCCode := 'C';
          end;
        end;
      end;
    end;
  end;

```

Begin

End.

This uses two additional variables.

- **LinePercentage:** This accumulates the ongoing value of Inventory read in
- **ABCCode:** This defines the ABC Code applicable to the current Line

Add these two fields to the Report in the MasterData1 Band. The final layout should look something like this

ABC Category Report							
Item Code	Description	Item Unit	On-Hand Qty	Unit Cost	Extended Cost	Cumulative	ABC
[MD_ 'ITEMCODE']	[MD_ 'ITEMDESCRIPTION']	[MD_ 'ITEM']	[MD_ 'ONHA']	[MD_ 'AVERAG']	[MD_ 'EXTEND']	[LinePercentage]	[ABC]
						[TotalValue]	
							Page

If you now run the Report each line will be allocated an ABC Code

'Save' the Report layout and go back to the 'Master Settings' screen

2.3. Adding Run Parameters

You have already added parameters earlier in this series of exercises. You can try adding them yourself now..... or go through the following steps

In the Master Settings screen add the following two parameters

Condition	Type
A to B Cutoff;{Integer}	9000
B to C Cutoff;{Integer}	9000

Click on the 'Edit' Button and enter the parameter values. For example:

A to B Cutoff: 60
B to C Cutoff: 85

In the Report Layout select **Report>Variables** and enter the Variables

A to B Cutoff
B to C Cutoff

Finally, click on the 'Code' tab and replace the current code with the following

```

Var
TotalValue, LinePercentage : Double;
ABCCode : String;

procedure TotalInventory(Sender: TfrxComponent);

```

```

begin
  if (engine.FinalPass = False) then
    begin
      TotalValue := TotalValue + (<MD_."ONHANDQTY"> * <MD_."AVERAGECOST">);

    End
  else
    begin
      LinePercentage := LinePercentage + (<MD_."ONHANDQTY"> *
<MD_."AVERAGECOST">);
    End;

    If ((LinePercentage / TotalValue * 100) < <A to B Cutoff>) then
      begin
        ABCCode := 'A';
      end
    else
      begin
        If (((LinePercentage / TotalValue * 100) >= <A to B Cutoff>) and
          ((LinePercentage / TotalValue * 100) < <B to C Cutoff>)) then
          begin
            ABCCode := 'B';
          end
        else
          begin
            ABCCode := 'C';
          end;
        end;
      end;
    end;

  Begin

  End.

```

If you now run the report you will see that it categorises the Items into A, B or C relative to your input parameters.

15.1.1.8.3 Updating Ostendo Database

This is an example of how you can update your Ostendo Database from within the Report and View developer. We will use the ABC evaluation routine in the previous exercise to update the database.

In the Master Settings screen add the following parameter

Condition	Type
Update Run;{Boolean}	9000

In the Report Layout select **Report>Variables** and enter the Variable **Update Run**

Finally, click on the 'Code' tab and replace the current code with the following

```

Var
  TotalValue, LinePercentage : Double;
  ABCCode, TheItemCode : String;

Procedure UpdateOstendo(Sender: TfrxComponent);

```

```

Begin
  if <Update Run> = 'Yes' then
    begin
      TheItemCode := <MD_."ITEMCODE">;
      UpdateDBValue('ITEMMASTER','ABCCCLASS',ABCCCode,'ITEMCODE',TheItemCode);
    End;
  end;

procedure TotalInventory(Sender: TfrxComponent);
begin
  if (engine.FinalPass = False) then
    begin
      TotalValue := TotalValue + (<MD_."ONHANDQTY"> * <MD_."AVERAGECOST">);

    End
  else
    begin
      LinePercentage := LinePercentage + (<MD_."ONHANDQTY"> *
      <MD_."AVERAGECOST">);
    End;

    If ((LinePercentage / TotalValue * 100) < <A to B Cutoff>) then
      begin
        ABCCCode := 'A';
      end
    else
      begin
        If (((LinePercentage / TotalValue * 100) >= <A to B Cutoff>) and
        ((LinePercentage / TotalValue * 100) < <B to C Cutoff>)) then
          begin
            ABCCCode := 'B';
          end
        else
          begin
            ABCCCode := 'C';
          end;
        end;
      end;
    end;

Begin

End.

```

If you now run the report and 'check' the '**Update Run**' parameter you will find that the ABC Code field in the Item Master record has been updated.

Note: The function '**UpdateDBValue**' is a standard function within the Report Writer and can be seen under the 'Functions' tab to the right of the Report Layout screen.

15.1.1.8.4 Select Report Formats

You can design multiple Report Layouts from a single data extract and select the specific Layout from the run conditions. This is currently used in Ostendo for Sales Quotes where you have the option to use a Standard Style Quote or a Letter Style Quote. In this exercise we will use the create (a) an Item Listing and (b) a List of Serial Numbered Items and their current Serial Numbers in stock.

1. Creating the Master Query

Create a new Report called (say) '**Item Codes**' and, in the 'Master Settings' screen define the following Master Query

```
Select * from ItemMaster
```

2. Creating the First Report Layout

Using the Report Wizard add **ITEMCODE**, **ITEMDESCRIPTION**, and **ITEMUNIT** to the report and click the '**Finish**' button. After ensuring that it will run via the '**Preview**' Icon save the Layout and return to the Master Settings panel.

3. Extending the Query

Click on the **Detail Queries** tab and against **Query #1** enter the following

```
Select * from INVENTORY where Itemcode = :Itemcode
```

Back to the '**Master Settings**' screen and click on the '**Edit**' button to go to the Report Layout.

4. Creating the Second Report Layout

Click on **Report>Data** and 'check' the checkbox against **DD_1** then 'save' the setting.

Click on **File>New Page** and create a new (blank) page.

Go back to '**Page 1**' and single-click on the '**Page Header**' Band. 'Right-mouse' and '**copy**'. Go to '**Page 2**' and '**Paste**'. Position at the top of the page.

Repeat the copy process for the '**MasterData**' Band

In **Page 2** add a '**DetailData**' Band and associate it with **DD_1**. Drag and drop fields **WAREHOUSECODE**, **LOCATIONCODE** and **SERIALNO** into the Detail Data band.

If you now **Preview** the Report it will run through Page 1 for all Items and then Page 2 for all Items and their Inventory records. What we now need to do is two things:

- Select which Page we wish to print out, and
- Restrict the Printout against Page 2 to Serial Numbered Items only.

4.1. Selecting the Page

Step 1. Go back to the 'Master Settings' screen and add the following Condition parameter:

```
Print Serial Numbers;{Boolean}          9000
```

Let's have a look at this in more detail

Print Serial Numbers: Open Format text that will be presented when running the Report to ask if you wish to run the Serial Number Page
; Separates the above Text from the next field
{Boolean} This is the format of this entry {contained in 'squiggly' brackets}. Data entry is validated against this format although the value itself is stored as a String in the database. The validation options are:

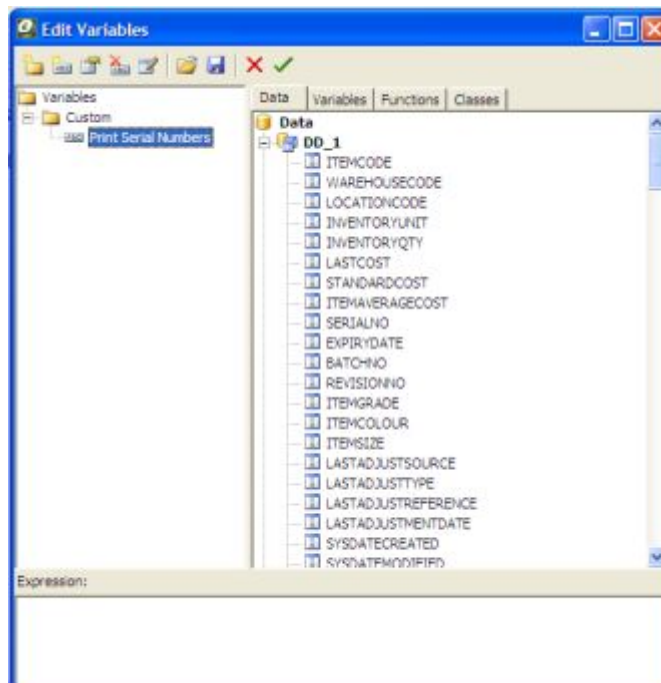
Boolean
Integer
String
Date
Time
Double

9000 The 'Type' of Parameter being entered. For User-defined entries this is always 9000

Step 2 Create a Report Variable

Go into the Report Layout and click on '**Report**' on the top toolbar and select '**Variables**' from the drop-down list. In the presented panel click on the '**Category**' icon (first icon on the top toolbar). This will create a new Variable called '**New Category 1**'. Single click on this and change the label to **Custom**. Click the Green 'Tick' icon to confirm the entry.

Having set up the Category select it then click on the '**Variable**' icon (second from the left on the top toolbar). This will create a new Variable called '**New Variable 1**'. Single click on this and change the label to **Print Serial Numbers**. The final screen should look like this



Click the **Green** 'Tick' icon to confirm the entry and exit the screen

Step 3 Create the Code

We now need to create some code to print the relevant Page depending upon the entry made against parameter '**Print Serial Numbers**'. Click on the '**Code**' tab and add the following

```
begin
  if (<Print Serial Numbers> = 'Yes') then
    begin
      Page1.visible := False;
      Page2.visible := True;
```

```

    end
  else
    begin
      Page1.visible := True;
      Page2.visible := False;
    end;
  End.

```

If you run the report and 'check' the Print Serial Numbers' parameter you will see that only Page 2 will be printed.

4.2. Restricting Page 2 to Serial Numbered Items

In the Report Layout click on the 'Code' tab and enter the following at the beginning of the current Code so that the full Code looks something like this

```

procedure SerNoOnly(Sender: TfrxComponent);
begin
  if (<MD_."SERIALTRACKED"> = 0) then
    begin
      DetailData1.visible := false;
      MasterData2.visible := false;
    end
  else
    begin
      DetailData1.visible := true;
      MasterData2.visible := true;
    end;
end;

begin
  if (<Print Serial Numbers> = 'Yes') then
    begin
      Page1.visible := False;
      Page2.visible := True;
    end
  else
    begin
      Page1.visible := True;
      Page2.visible := False;
    end;
end;
End.

```

Go back to the 'Page 2' tab and single click on the 'Master Data' Band'. Click on the 'Event' tab (down the left of the screen) and - against **OnBeforePrint** - select your procedure 'SerNoOnly'.

If you go back to the 'Master Settings' screen you can Preview the report where you should be able to selectively print Page 1 or Page 2

15.1.1.8.5 User-Defined Variables

Taking the previous Report a step further you can declare Variables and carry out ongoing calculations, etc. These can be used as controls or accumulators within the Report. The ongoing or final value of this variable can be printed on the report itself.

We will create a variable that checks if the Item being read is Serial Controlled and, if so,

increment the Variable. At the end of the report we will print the Variable to show the number of Items that are Serial controlled.

Click on the **Code** tab and add this variable to the beginning of the Code

```
Var
  SerNoCount: Integer;
```

The next step is to increment the variable whenever an printed Item is found to be Serial Controlled.. This has two elements

- Create a Procedure to increment the Variable
- Tell the Report when to run the Procedure (On before Print)

The procedure would look something like this:

```
procedure SerCountOnBeforePrint(Sender: TfrxComponent);
begin
  if (<MD_."SERIALTRACKED"> = 'True') then
    begin
      SerNoCount := SerNoCount + 1
    end
end
End;
```

The full Code should then be

```
Var
  SerNoCount: Integer;

procedure SerCountOnBeforePrint(Sender: TfrxComponent);
begin
  if (<MD_."SERIALTRACKED"> = 'True') then
    begin
      SerNoCount := SerNoCount + 1
    end
end
End;

procedure SerNoOnly(Sender: TfrxComponent);
begin
  if (<MD_."SERIALTRACKED"> = 0) then
    begin
      DetailData1.visible := false;
      MasterData2.visible := false;
    end
  else
    begin
      DetailData1.visible := true;
      MasterData2.visible := true;
    end;
end;

begin
  if (<Print Serial Numbers> = 'Yes') then
    begin
      Page1.visible := False;
      Page2.visible := True;
```

```

end
else
begin
Page1.visible := True;
Page2.visible := False;
end;

begin
if (<Print Serial Numbers> = 'Yes') then
begin
Page1.visible := False;
Page2.visible := True;
end
else
begin
Page1.visible := True;
Page2.visible := False;
end;
end;
End.

```

To tell the Report when to run the Procedure go into the '**Page1**' Report Layout and single click on the '**MasterData1**' Band to select the band. Down the left of the screen click on the 'Events' tab and - against '**OnBeforePrint**' - select Procedure '**SerCountOnBeforePrint**'

Now that we have added the variable and incremented it as required we now need to print out the variable at the end of the Report

Click on the '**Insert Band**' Object down the left of the screen and select '**Report Summary**'. This will add a band at the bottom of **Page1**. Now add a '**Text Object**' into the band. In the presented '**Text**' panel add the following

Number of Serial Items is [SerNoCount]

This is a combination of open-format text plus your Variable. You should note that the Variable should be enclosed in 'Square' Brackets. This tells the Report Writer that it is a Variable.

If you now run the Report you may find that it shows twice the number of Serial Items you expected. This is because you are still running a 'Two-Pass' report and the Variable is incremented through each pass. Therefore go to **Report>Options** and 'uncheck' the '**Double Pass**' checkbox.

15.1.1.8.6 Run Ostendo Action from Preview

You can click a field in the report 'Preview' (I.e. displayed on your screen) to carry out an activity. In this example we will use the '**Item Code**' on '**Page1**' of the Report you created in the previous exercise and open Ostendo's Item Master screen for the selected record.

Go into the Report Layout for the report you created in the previous exercise and:

Step 1. Click on the '**Code**' tab down the left of the layout and add the following Procedure:

```

Procedure ItemOnPreviewClick(Sender: TfrxView; Button: TMouseButton; Shift: Integer;
var Modified: Boolean);
Var
  TheItem: String;

```

Begin

```
TheItem := TfrxMemoView(Sender).Text;
RunSystemAction('Inventory','Items','Itemcode',TheItem,1);
```

End;

Note: The parameters associated with **RunSystemAction** can be seen by clicking on the 'Functions' tab to the right of the screen and going to the '**Advanced**' section

Step 2. Click on '**Page1**' tab.

In that layout carry out the following.

- Single-click on the **[MD_."ITEMCODE"]** object in the **MasterData1** Band. This selects the Object
- Click on the '**Events**' tab down the left of the screen. In the field adjacent to '**OnPreviewClick**' select '**ItemOnPreviewClick**' from the drop-down list.

That's it !!

Save the changes and exit the Report Layout to take you back to the '**Master Settings**' screen. Click on the '**Preview**' button and click the '**OK**' button (Don't check the '**Print Serial Numbers**' checkbox as you wish to run **Page1** of the report).

Return the Report to your screen by clicking on the '**Screen**' Icon.

In the presented Report click on any Item Code. You will find that Ostendo's Item Master Screen will appear.

15.1.1.8.7 Restrict Printing to Specific Users

You can restrict who can, and cannot, print a Report by going into the Report Layout and clicking on the '**Code**' tab and entering the following before the final **end.** statement. In this example we are denying User 'ADMIN' from printing this document

Begin

```
if uppercase(CurrentUser) = 'ADMIN' then
  Showmessage('You are not allowed to print this document');
  Terminate;
```

end;

15.1.1.9 Other Report actions

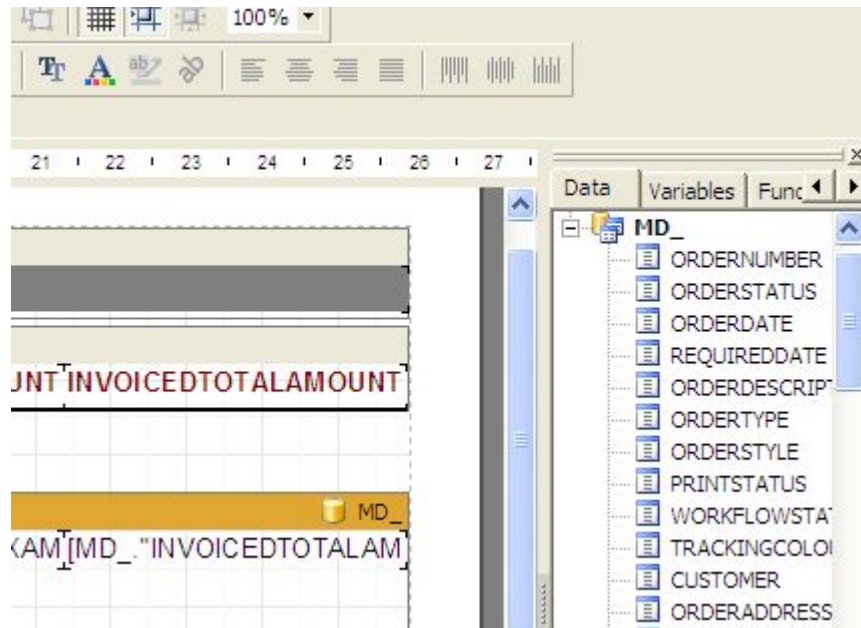
Here is a summary of some activities that can be carried out in the Report Layout

15.1.1.9.1 Creating a Data Field

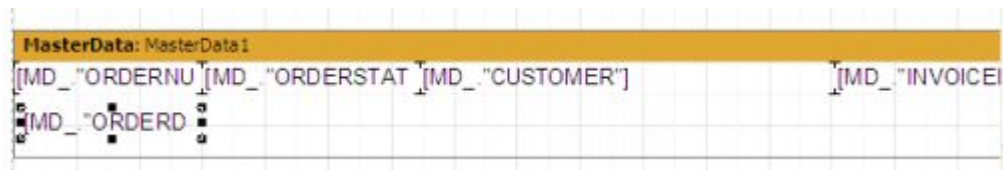
We will add the Sales Order Date to the report. As there may not be enough room in the current single line you should click on the bottom edge of the '**MasterData: MasterData1**' Band and drag it downwards so that it will accommodate another line of data.



You will see, down the right hand side, a list of all the fields in the Sales Order Header. If you cannot see this then click on **View>Toolbars>Data Tree**



Click on field 'ORDERDATE' and drag it into the band

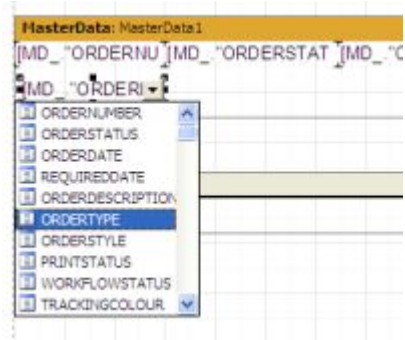



If you now click on the **'Preview'**  Icon you can see the finished listing

15.1.1.9.2 Changing a Data Field

We will change the field that you added in the previous example from INVOICEDATE to ORDERTYPE

Click on the field INVOICEDATE. Then click on the down arrow to the right of the field to display the available fields and select ORDERTYPE

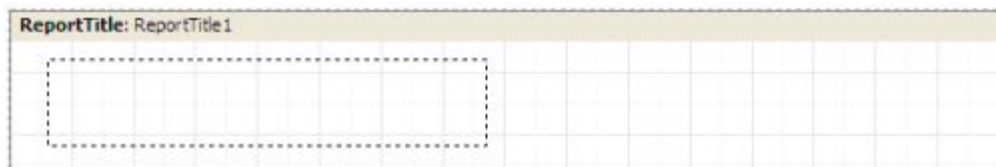


If you now click on the 'Preview'  Icon you can see the amended listing

15.1.1.9.3 Creating a Text Field

This exercise shows you how to add a Text (non-data related) field to the Report. You would use this type of field for describing data fields, Report Titles, permanent Descriptive information such as Terms and Conditions), etc

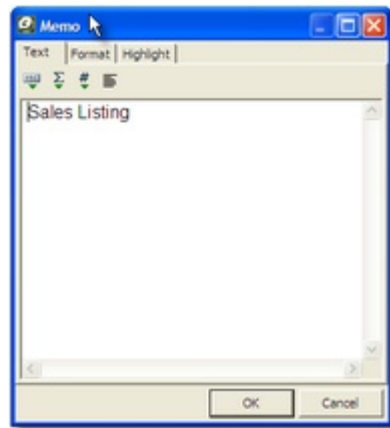
On the Toolbar click on the 'Text Object' Icon  then move the cursor to the report and place the displayed rectangle into the lower part of a Band as shown below



The following panel will appear

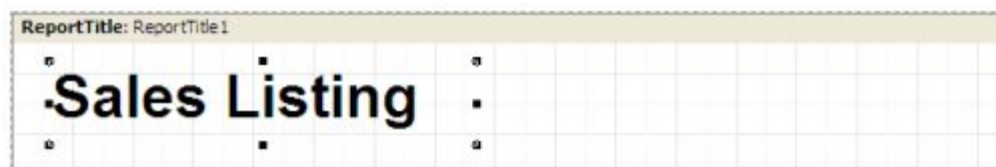


Simply Key in the Text and then click the **OK** button



You can use other tools to enhance the result such as

- Increasing the Font size
- Printing in 'Bold'
- Changing the Colour
- Adding a separator line (using the 'Line Object' in the left toolbar)
- Increasing the depth of the Report Title Band




Run the report in 'Preview' mode.

15.1.1.9.4 Adding a Date to a Report

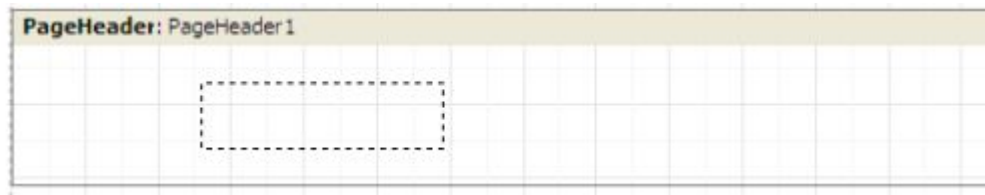
A date can be added to print out:-

- Once on a Report at the beginning of the report
- On each page in the Page Header and/or Page Footer

In this exercise we will add a Date to a Page Header. Therefore add a Page Header Band by


clicking on the **'Insert Band'** Icon  and selecting **'Page Header'** from the band options

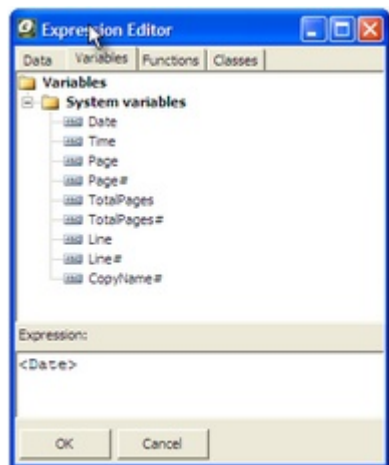
On the Toolbar click on the 'Text Object' Icon  then move the cursor to the report and place the displayed rectangle into the lower part of the Band as shown below



The following panel will appear



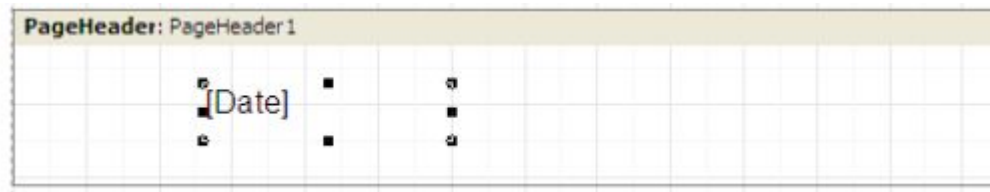
On the displayed 'Memo' panel click on the Insert Expression icon  to display the Expression Editor. Select the 'Variables' Tab and 'double click' the mouse on 'Date' within that display.



Click the 'OK' button to return to the 'Memo' panel



The 'Memo' screen will be filled with **[DATE]**. Of course you could have keyed in **[DATE]** directly rather than go to the Expression Editor. Click the **OK** button to confirm the entry and return to the Report.



Run the report in 'Preview' mode.

Of course you would also need to indicate that this is a date so return to the Memo panel and type in the required Text.




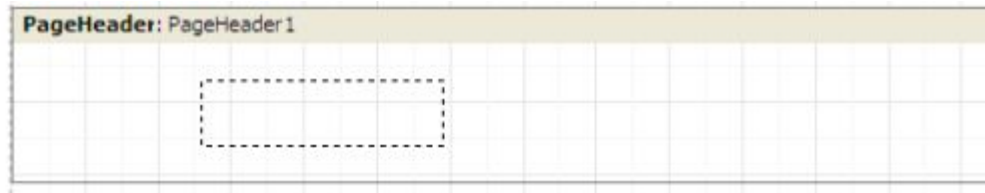
Again, run the report in 'Preview' mode.

15.1.1.9.5 Adding Page Numbers

A Page Number can be added to print out:-


- On each page in the Page Header
- On each page in the Page Footer

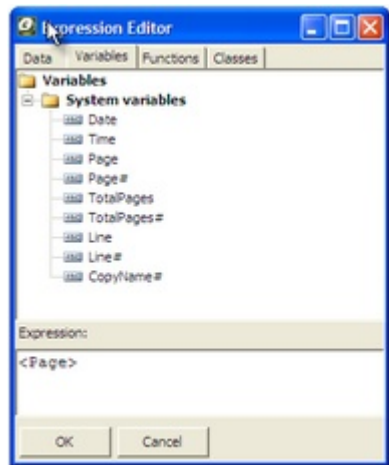
On the Toolbar click on the 'Text Object' Icon  then move the cursor to the report and place the displayed rectangle into the lower part of the Band as shown below



The following panel will appear



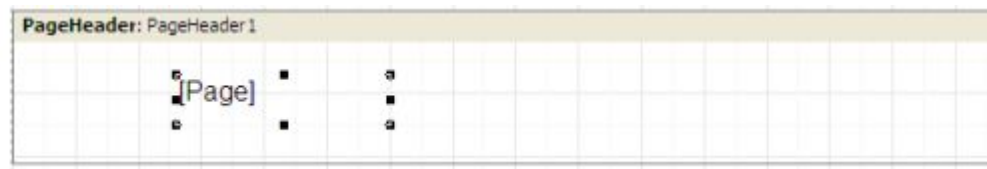
On the displayed 'Memo' panel click on the Insert Expression icon  to display the Expression Editor. Select the 'Variables' Tab and 'double click' the mouse on 'Page' within that display.



Click the 'OK' button to return to the 'Memo' panel

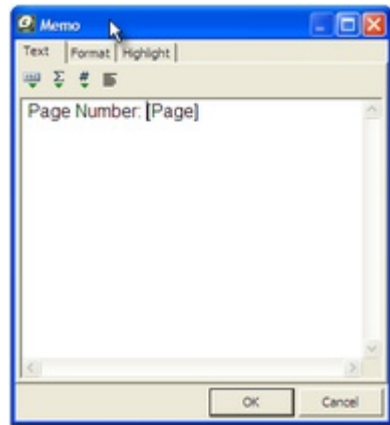


The 'Memo' screen will be filled with **[PAGE]**. Of course you could have keyed in **[PAGE]** directly rather than go to the Expression Editor. Click the **OK** button to confirm the entry and return to the Report.



Run the report in 'Preview' mode.

Of course you would also need to indicate that this is a Page Number so return to the Memo panel and type in the required Text.



Again, run the report in 'Preview' mode.

15.1.1.9.6 Adding a Report Title

A Report Title appears once at the beginning of a Report. This can be on its own page or on the first page prior to the data being printed. For this second option see section on **'Creating a Text field'**

If you wish to place the Title on its own page at the beginning of the report then carry out the

following.

On the outside of the current Report Page you will see a 'Tab' with the title 'Page1'


Click Right Mouse on this and select 'Add Page' to create a new page.



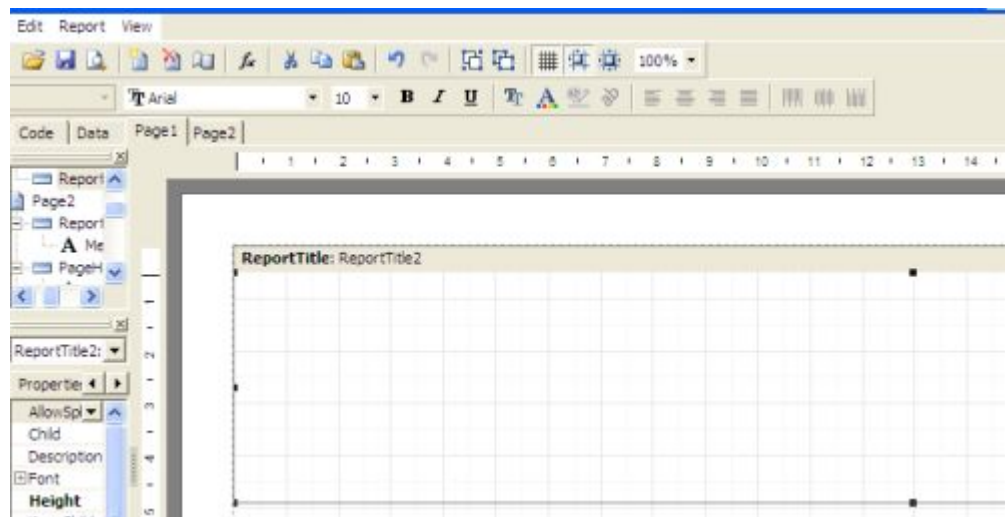
You now need to move the data from **Page 1** to the new **Page 2**. therefore:-

- Go back to Page 1 and select **Edit>Select All**
- Select **Edit>Cut** then got to Page 2
- Select **Edit>Paste**
- Drag and drop the copied information.

Return to Page 1 and carry out the following

Click on the 'Insert Band' Icon  and select the **Report Title** Band from the drop-down list.

The displayed Band has two portions. The upper portion tells you that it is a Report Title Band and the lower portion is where you place the Title. If you click on the Band then eight 'Handles' will appear. You can drag and drop these handles to increase or decrease the space occupied by the band in the finished report.



Insert a **Text** Object into this Band and format the Text as described in an earlier exercise

15.1.1.9.7 Generating Sub Totals

Sub Totals can be added to a Report at various Levels such as

- Footer
- Group Footer
- Page Footer
- Report Summary

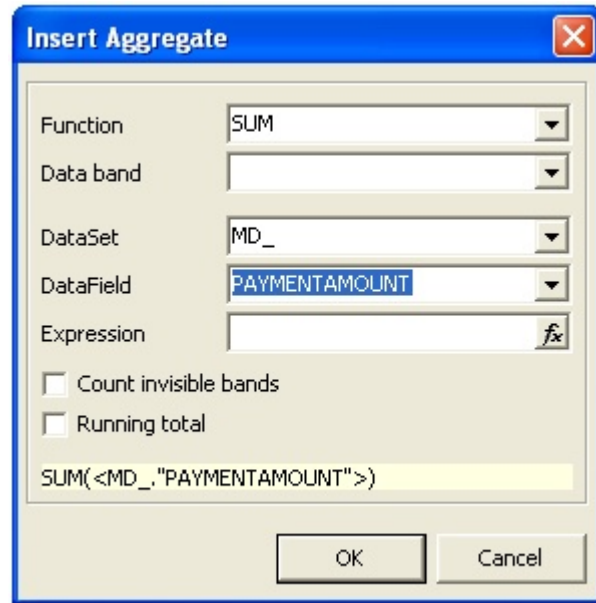
The method by which this achieved is the same in each case with the field appearing in the relevant Reporting Band that informs the program when to generate and print the Sub Total

To create a Sub-Total field carry out the following steps:-

Create a Text Object as described in an earlier exercise and place it in one of the summary bands (Footer, Group Footer, Page Footer, Report Summary)



On the displayed 'Memo' panel click on the  icon to display the **'Insert Aggregate'** panel



Select the function '**SUM**' from the drop-down list
Select the Dataset where the field resides
Select the Field that is being 'Summed'
Click the '**OK**' button when done



Click **OK** to complete the task


Hint:- If you know the format of the expression you can simply enter **[SUM(<MD_,"FREIGHTTOTALAMOUNT">)]** directly into the 'Memo' Panel.

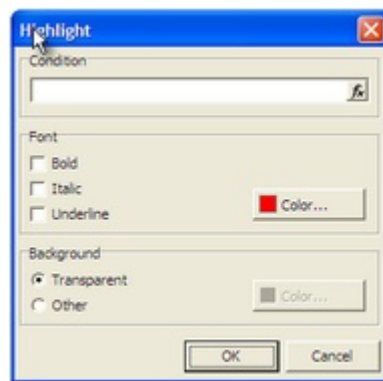
15.1.1.9.8 Creating Conditional Highlighting

With this feature you can specify a condition of a field (Example:- Value > 5000, or Invoice Date < a specified Date) and, depending upon the condition, change the display properties of the field (for example Bold, Red Text, or Green background to the field).

In this example we will highlight where the Counted Quantity is not equal to the Book Quantity in the Inventory Count Listing Report. Therefore copy the Inventory Count Sheet to your Company Folder

Click on field [DD_1."COUNTQTY"] in the Detail Data Band

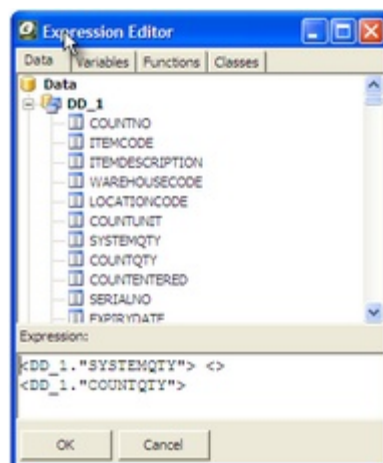
Select the 'Conditional Highlighting' Icon  found on the 'Text' Toolbar



On the displayed 'Highlight' panel click on the **fx** button and define the condition. In our example we wish to highlight where [DD_1."COUNTQTY"] does not equal [DD_1."SYSTEMQTY"]

'Double Click' on the COUNTQTY field which will place the selected field in the lower panel. Define the condition to be satisfied to create the highlight. In this example <> (Not equal)
'Double Click' on the SYSTEMQTY field which will place the selected field in the lower panel.

The resultant selection should appear like this



Click 'OK' to return to the 'Memo' panel and define the highlight itself

Select the Font style and/or the background colour if this condition is satisfied.

Click 'OK' to close.

If you run the report then the following printout will be produced with the relevant Count fields highlighted in selected colour

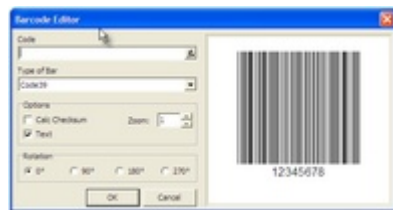
Reference	Date	Status			
Highlight where Count not equal Book Stock	03 Sep 2008	InProgress			
Location	Item Code	Unit	System Qty	Count Qty	Counted
Warehouse: Main					
Primary	100-2002	Each	1999	1999	<input checked="" type="checkbox"/>
Washer-Mild Steel-10mm					
Primary	100-2004	Each	1990	1989	<input checked="" type="checkbox"/>
Washer-Mild Steel-12mm					
Primary	100-2006	Each	2000	2000	<input checked="" type="checkbox"/>
Washer-Mild Steel-14mm					
Primary	100-2008	Each	2000	2000	<input checked="" type="checkbox"/>
Washer-Mild Steel-16mm					
Primary	110-2035	Each	2000	2000	<input checked="" type="checkbox"/>
Washer-Stainless Steel-10mm					
Primary	110-2037	Each	1948	1947	<input checked="" type="checkbox"/>

15.1.1.9.9 Adding Barcodes

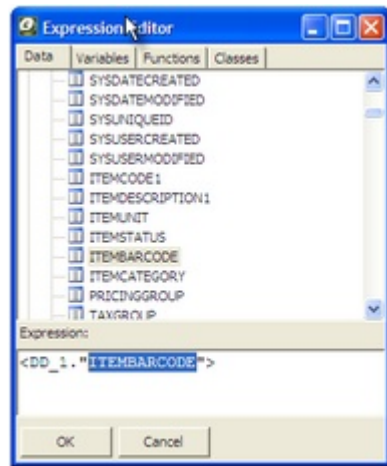
You can print any Text or Database field onto the document as a barcode field. In this example we will add the barcode of the Item Number to the Inventory Count Listing Report. If you haven't done the previous exercise (Highlighting) then copy the Inventory Count Sheet to your Company Folder



Select the 'Barcode Object' icon and move the cursor into the Detail Data Band of the Report and drop it where you wish the Barcode to be printed. A Barcode Editor panel will appear.



You may type fixed text into the 'Code' field or click on the **fx** drop-down button to display the data fields. In our exercise double click on field **ITEMBARCODE** then click the 'OK' button



Choose the following


- Type of Bar from the drop-down list
- If it contains a checksum
- Print the size
- Identify its rotation
- Click 'OK' when done

The result should look something like this

INVENTORY LISTING				Development-X Limited	
Grouped by Product					
Item Code	Description	Unit	Qty on Hand		
00-ASSEMBLIEDITM	Assembled Item Test	Each	240		
Warehouse	Location	Qty	Unit	00-ASSEMBLIEDITEM	
Bulk	Primary	6	Carton		
Bulk	Primary	2	Each		
Main	Primary	13,0833	Carton		
Main	Primary	9	Each		
00-BATCHTEST	Testing Batch Controlled Item	Each	115		
Warehouse	Location	Qty	Unit	00-BATCHTEST	
Main	Primary	100	Each	Batch: 1000	
00-COLOURTEST	Testing Colour Controlled Item	Each	-31		
Warehouse	Location	Qty	Unit	00-COLOURTEST	

15.1.1.9.10 Adding Lines and Shapes

You can add Lines of various formats, Rectangles, Triangles, Ovals, etc to the Report

Select the 'Draw' icon  then select the shape from the drop-down list and place the cursor on the main area and click on the left mouse. Depending upon the selected shape you can manipulate the size and aspect ratio as required. For rectangles, etc the created (blank) shape will be surrounded by 8 dots (known as 'handles'). Select one of the handles to expand or contract

the shape. If you select the centre of any line or shape and hold down the left mouse you can move the object around the page

Formatting Options

If you click on the created object then the following options are available to you on the 'Frame' Toolbar



Select the colour to fill the shape



Select the colour of the line around the shape



Select the style of the line around the shape



Select the thickness of the line around the shape

15.1.1.9.11 Creating a Multiple Column Report

You can identify that a Report has a multi-column output by simply clicking on **File>Page Settings** and then clicking the 'Other Options' tab and defining the number of columns in the page

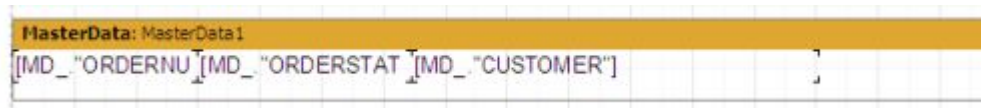
15.1.1.9.12 Adding a Record Count

This feature allows you to count the number of records contained in a 'Group' or Report. Your report should have the following Bands

- Master Data
- Group Footer and/or Report Summary

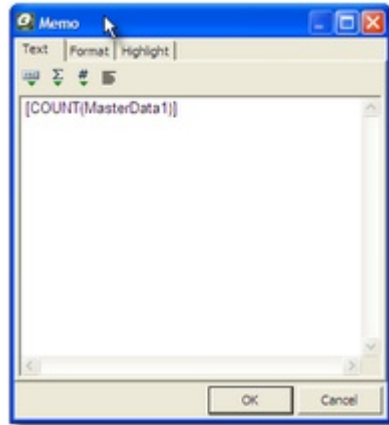
To display the counted records carry out the following steps:-

Step 1. to the right of the Master Data Band name you will find the Identity of the Band



In this example it would be **MasterData1**

Step 2. Insert a Text Object **A** into the 'Group Footer' and/or the 'Report Summary' Band.



In the displayed Memo Panel enter `[[COUNT(MasterData1)]]`. Click the **OK** Button to confirm

If you now run the report you see the number of counted records at the end.

Of course you would also need to indicate what the printed number represents so return to the Memo panel and type in the required Text.



Again, run the report in 'Preview' mode.

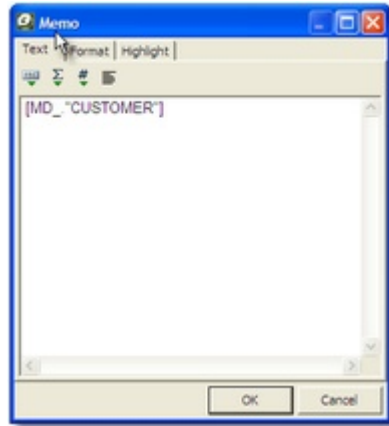
15.1.1.9.13 Printing part of a Field


In this example let us address the following report format

MasterData: MasterData1		
[MD_"ORDERNU	[MD_"ORDERSTAT	[MD_"CUSTOMER"]

and only print out the first eight characters of the 'Customer'

Double Click on the MD_CUSTOMER field to bring up the 'Memo' panel



Click on the Expression Editor  icon and then the 'Functions' tab. On this screen scroll down to the 'String' section and double click on

Copy(s: String; from, count, Integer): String

You will see **Copy(, ,)** displayed in the bottom panel.

You now need to complete this entry to produce:

Copy(<MD_."CUSTOMER">,1,6)

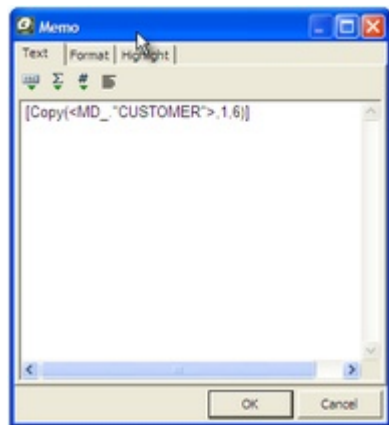
Where <MD_CUSTOMER"> is the field being referenced

1 = The character number in the field where the report is to start

6 = The number of characters - from the start character - to print

Hint: You can place the cursor between (and the first comma and then select the 'Data' tab then double click on 'CUSTOMER'

Click the 'OK' button to return to the 'Memo' screen, which now shows



If you 'Preview' the Report you will see that the Customer Field is restricted to 6 characters

15.1.1.9.14 Printing a Linked Image

This exercise will take you through creating a Report that will use the Item Master File and will show you how to print linked Images from the Image Master File.

1. Preparation

You should go into *Inventory>Items* and select the Item Number that is to have the linked Image. Click on the 'Related' Button and select 'Item Images' to add an image to the Item. Note: You can alternatively go to *General>Image Management* and create the link.

2. Creating the Report

Go into *File>Reporting Configuration>Report and View Developer*

2.1. Creating the Selection Criteria

Click the 'Add' button and create a new report (say) 'Item Image Test'

Complete the fields in the upper part of the screen as follows

Include in Main Menu: 'Checked'
Name: Prefilled with 'Item Image Test'
Manu Order: Enter where you want it to appear under Ostendo Inventory 'Reports'
Category: Select 'Inventory' from the drop-down list
Type: Prefilled with 'Report'
Specific Screen: Leave Blank
Report File Name: Prefilled with 'Item Image Test'

In the 'Master Query' area enter the following

```
select * from ITEMMASTER order by ITEMCODE
```

In the 'Condition' area enter the parameter

```
Item Code;ITEMCODE=:SCREENPARAM  
Type = 1004
```

Click on the 'Detail Queries' tab and enter the following in Query#1

```
select * from IMAGEMASTER where IMAGETYPE =  
'Item' and LINKEDTO = :ITEMCODE  
order by IMAGENAME
```

This provides a Parent (Item Number) to Child (Image) link with a selection parameter that will allow you to select a specific Item Number.


2.2. Creating the Report


Go into the 'Master Settings' tab click on the 'Edit' button (to the right of the screen). A panel will appear for you to select the Item Number. Select the above Item and click the OK button.

On the presented panel select 'Standard Report Wizard' and click OK

Go to the 'Fields' tab and move **ItemCode** and **ItemDescription** to the right panel. Click the '**Finish**' button

On the presented layout click on **Report>Data** on the top toolbar and 'check' the DD_1 checkbox.

Click on the 'Insert Band' icon  and select 'Detail Data'. A 'Detail Data' Band will be inserted into the Report. In the presented panel select DD_1. Expand the band in preparation for inserting the picture.

Click on the 'Picture Object' button  (down the left of the screen) and position it in the 'Detail Band' then click the Green 'tick' symbol. Resize the Picture and/or Band as required.

2.3. Linking the Picture object to the Item

For the next step you need to have the '**Object Inspector**' on your screen. To display this go to **Views** on the top toolbar then select '**Toolbars**' and then '**Object Inspector**'

- With the above 'Picture Object' selected you will see (on the left) that the field name is **Picture1: TfrxPictureView**. Under that are two tabs. Select the '**Properties**' tab and 'check' the 'Stretched' checkbox and uncheck the 'Autosize' checkbox

- Now select the DetailData1 Band and you will see (on the left) that the field name is **DetailData1: TfrxDetailData**. Under that are two tabs. Select the '**Events**' tab and in the field '**OnBeforePrint**' enter **DetailData1OnBeforePrint**

Finally, above the **DetailData1: TfrxDetailData** you will see a tab called '**Code**'. In right hand panel enter the following

Click on the '**Code**' tab (To the left of the screen) and enter the following code.

```

procedure DetailData1OnBeforePrint(Sender: TfrxComponent);
begin
  if (<DD_1."IMAGEFILE"> <> null) and (trim(<DD_1."IMAGEFILE">) <> "") then
    begin
      DetailData1.visible := True;
      Picture1.loadfromfile(<DD_1."IMAGEFILE">);
    end
  else
    begin
      DetailData1.visible := False;
    end;
  end;
begin
end.

```

If you now click on the 'Preview' Icon on the top Icon Bar you should see the finished report.

2.4. Items without an Image

The Report as it stands will only print those Items that contain an Image. It is most probable that

you will also want to print those Items that don't have an Image attached to them. To achieve this go into the Report Layout Edit screen and click on the 'MasterData1' Band. Now refer to the Object Inspector (Properties) panel. You will see a field called 'PrintIfDetailEmpty'. 'Check' that field.

15.1.1.9.15 Printing Items with linked Properties

This exercise will take you through creating a Report that will use the Item Master File and will show you how to print linked Properties from the Properties Master File.

1. Preparation

You should go into *Inventory>Items* and select the Item Number that is to have the Properties. Click on the 'Related' Button and select 'Item Properties' to add Properties to the Item.

2. Creating the Report

Go into *File>Reporting Configuration>Report and View Developer*

2.1. Creating the Selection Criteria

Click the 'Add' button and create a new report (say) '**Item Properties Test**'

Complete the fields in the upper part of the screen as follows

Include in Main Menu: 'Checked'
Name: Prefilled with 'Item Properties Test'
Manu Order: Enter where you want it to appear under Ostendo Inventory 'Reports'
Category: Select 'Inventory' from the drop-down list
Type: Prefilled with 'Report'
Specific Screen: Leave Blank
Report File Name: Prefilled with 'Item Properties Test'

In the '**Master Query**' area enter the following

```
select * from ITEMMASTER order by ITEMCODE
```

In the '**Condition**' area enter the parameter

```
Item Code;ITEMCODE=:SCREENPARAM  
Type = 1004
```

Click on the '**Detail Queries**' tab and enter the following in Query#1

```
select * from ITEMPROPERTIES where ITEMCODE = :ITEMCODE  
order by PROPERTYNAME
```

The above now provides a Parent (Item Number) to Child (Properties) link with a selection parameter that will allow you to select a specific Item Number.

2.2. Creating the Report

Go into the '**Master Settings**' tab click on the 'Edit' button (to the right of the screen). A panel will appear for you to select the Item Number. Select the above Item and click the OK button.

On the presented panel select '**Standard Report Wizard**' and click OK

Go to the 'Fields' tab and move ItemCode and ItemDescription to the right panel. Click the 'Finish' button

On the presented layout click on **Report>Data** on the top toolbar and 'check' the DD_1 checkbox.



Click on the 'Insert Band' icon and select 'Detail Data'. A 'Detail Data' Band will be inserted into the Report. In the presented panel select DD_1.

For the next step you need to have the 'Data Tree' on your screen. To display this go to **Views** on the top toolbar then select '**Toolbars**' and then '**Data Tree**'. The fields should now display on the right of your screen. If the field List locate DD_1 and the Properties fields will be displayed. Drag and drop the required fields into the Detail Data Band.

If you now click on the 'Preview' Icon on the top Icon Bar you should see the finished report.

2.3. Items without an Property

The Report as it stands will only print those Items that contain Properties. It is most probable that you will also want to print those Items that don't have Properties attached to them. To achieve this go into the Report Layout Edit screen and click on the 'MasterData1' Band. Now refer to the Object Inspector (Properties) panel. You will see a field called '**PrintIfDetailEmpty**'. 'Check' that field.

15.1.1.9.16 Selective printing within the report

In this exercise we will select all Items via the Master Query but restrict the printout to those Items that have negative Stock

1. Creating the Report

Go into **File>Reporting Configuration>Report and View Developer** and click the 'Add' button and create a new report (say) '**Negative Stock Items**'

Complete the fields in the upper part of the screen as follows

Include in Main Menu: 'Checked'

Name: Prefilled with 'Negative Stock Items'

Manu Order: Enter where you want it to appear under Ostendo Inventory 'Reports'

Category: Select 'Inventory' from the drop-down list

Type: Prefilled with 'Report'

Specific Screen: Leave Blank

Report File Name: Prefilled with 'Negative Stock Items'

In the '**Master Query**' area enter the following

```
select * from ITEMMASTER order by ITEMCODE
```

Click on the '**Edit**' button (to the right of the screen). On the presented panel select '**Standard Report Wizard**' and click OK

Go to the '**Fields**' tab and move ItemCode, ItemDescription, ItemUnit, and OnHnadQty to the right

panel. Click the **Finish** button to generate the Report

Click on the **Code** tab to the left of the Report Edit panel and add the following

```
procedure MasterData1OnBeforePrint(Sender: TfrxComponent);
begin
  if (<MD_."ONHANDQTY"> < 0)
  then
  begin
    MasterData1.visible := True;
  end
  else
  begin
    MasterData1.visible := False;
  end;
end;

begin

end.
```

- Now select the **MasterData1** Band and you will see (on the left) that the field name is **MasterData1: TfrxMasterData**. Under that are two tabs. Select the **Events** tab and in the field **OnBeforePrint** enter **MasterData1OnBeforePrint**

If you now click on the 'Preview' Icon on the top Icon Bar you should see the finished report without negative Receipts.

15.2 Analysis Views

'Views' in Ostendo provides facility for Users to call up complete sets of data from all major sections of Ostendo, view and analyses that information, and output to various media.

In these exercises we will:

- Create a new View with data from the Customer records
- Look at some of the actions that can be carried out with a View
- Merge the displayed Data with Microsoft Word

15.2.1 Creating an Analysis View

Go into the Report and View Developer via **File>Reporting Configuration>Report and View Developer**.

You will see all the Reports, Views, Charts and Pivot Views that are supplied with Ostendo (These are held at the 'System' Level). Any new Report is automatically created with a Level of 'Company'

Click on the 'Add' button and, in the presented panel enter the following:

- 'Check' the **Create from Scratch** Radio Button
- 'Check' the **Analysis** Radio Button
- Name: enter **Customer Analysis**
- Click the **Create** button

You will be taken to the **Master Settings** tab of the Report layout.

Step 1. Naming the View

In the upper part of the screen enter the following:

- **Include in Main Menu** is 'ticked' to denote that it will appear in the Ostendo menu
- **Name:** Leave as '**Customer Analysis**'
- **Menu Order:** enter a number to denote where it appears in the '**Views**' Menu
- **Category:** From the drop-down list select '**Sales**'
- **Master Key Field:** Leave blank
- **Detail Key Field:** Leave blank
- **Title:** '**Customer Analysis**'
- **Merge Word Document:** Leave blank
- **Merge Data File:** Leave blank

Step 2. Defining the Data

The next step is to tell Ostendo from which table and what data is to be extracted. For our exercise we will extract all data from the Customer Master table. It is not the objective here to go through how to create Queries as this is covered in a separate document.

Click on the '**SQL Builder**' button to the right of the '**Master Query**' panel. In the presented screen locate '**CUSTOMERMASTER**' in the '**Tables**' panel. (Hint: type C to take you straight to C in the list). 'Double Click' on this and you will see '**Select from CUSTOMERMASTER**' appear in the '**Query**' panel with the cursor located after the word 'Select'. Key in an asterisk so that the statement will be '**Select * from CUSTOMERMASTER**'. This is telling Ostendo to select all fields from the Customer Master table.

Click the 'OK' button to exit the SQL Builder screen

Step 3. Test the Analysis View

Click on the '**Preview**' button to the right of the screen to see the results

Viewing the new View

If you exit the Report and View Developer and then go to **Sales>Views** you will see your new View in the list. If you select the View then the full Customer Details will be displayed. You can manipulate the details of that screen as outlined in the following exercises

15.2.2 Working the Analysis View

The main Analysis screen shows the retrieved information from your database. The extracted records can be 'sliced and diced' in the following manner.

Search & Sort

The following features are available that allow you to sort and filter the displayed data.

- sort any column in the displayed records into ascending or descending sequence by clicking on the selected column heading
- you may 'filter' the records based upon your own selection criteria by clicking on the blue triangular symbol in the selected column heading. From the drop-down list you may select the following:-
 - All - displays all records

- Blanks - Displays only those records that have no data in the field
- Non-Blanks - Displays only those records that contain data in this field
- Select the specific field content

Moving and Hiding Columns

You may also move columns as well as take unwanted columns from the display

- You can move columns by clicking on the column heading and 'dragging' the column into the required position.
- If you double click on the Column Heading's right edge you can make the column automatically 'close up' to match the amount of data in the field. You can also achieve this by 'Right Clicking' the column heading and selecting 'Best Fit'.
- If you wish to 'Close Up' all the columns then you should 'Right Click' on any column heading and select 'Best Fit - All Columns'.
- If you 'Right Click' on a column Heading and select 'Field Chooser' then a panel will appear for storing unwanted columns and enables you to 'Customise' the screen. You can move any unwanted columns to this panel by simply clicking on the column heading and dragging the column into this panel. You can recall stored columns by reversing this procedure.

An alternative to this is to 'Right Click' on a column Heading and select 'Remove this column'. This will automatically place the unwanted column in the storage panel. To hide the 'Field Chooser' Panel simply click on the 'x' in the upper right corner of the panel

Grouping

You may wish to Group 'like' records. To do this simply drag the required column heading into the area at the top of the screen where it states 'Drag a column header here to group by that column'. (If this area is not visible then you should 'Right Click' on any column heading and select 'Group By Box')

The screen will now group all records where the content of the 'Grouped' field is the same.

- If you click on the '+' indicator against each Group you can see the detailed records.
- This Grouping facility is not just single level. You can Group within Group, etc by simply dragging and dropping the 'sub-group' to the right of the first Group.
- This can be repeated for as many levels as you require.
- An alternative method of achieving this is to 'Right Click' on the selected column heading and select 'Group By This Field')

Summary Totals

For each Group you may wish to display summary totals. To do this, expand a Group by clicking on the '+' indicator to display the individual records. At the bottom of the Group List is a blank area. If this area is not visible then 'Right Click' on a column heading and select 'Group Footers'.

Now go to this blank area under any column and 'right click' the mouse to display the following options. These can be accessed depending upon the type of field (For example, you cannot 'Sum' a Date field)

- | | |
|-----------|------------------------------|
| ▪ Sum | Numeric fields only |
| ▪ Min | Numeric and Date fields only |
| ▪ Max | Numeric and Date fields only |
| ▪ Count | All fields |
| ▪ Average | Numeric fields only |
| ▪ None | All fields |

The selected column will now display the summary information

Grand Totals

You may also wish to display grand totals for the displayed data. To do this you should see a blank area at the end of the displayed list. If this area is not visible then 'Right Click' on any column Heading and select 'Footer'.

As with Group Footers, you can go to the blank area under any column and 'right click' the mouse to display the options.

15.2.3 Merging data with Microsoft Word

You have the option to print a document from within Ostendo whereby 'User-defined' source data can be merged with a Word document to produce printed forms with data from both sources

In this exercise we will create a letter to be sent to certain Suppliers. The Letter contains a combination of data fields from Ostendo and user Text. In our example we will produce the following letter

To <Company Name>
<Address Line 1>
<Address Line 2>
<Address Line 3>

Dear <Primary Contact>

This is my test document for merging.

Regards

Step 1. Extract the Supplier Data

Go into **File>Reporting Configuration>Report and View Developer**. Click the 'Add' button and, in the creation screen, 'check' the 'Analysis' Radio Button and enter 'Suppliers' in the 'Name' field

On the presented 'Master Settings' tab:

- Select 'Purchasing' from the drop-down against **Category**
- Against 'Merge Word Document' enter a document name (Example: **SuppLetter**) that will become the Merge Template Document. Note: use the 3 dots to generate the **full** path
- Against 'Merge Data File' enter a file name (Example: **SuppSource**) that will become the Merge Source Data. Note: use the 3 dots to generate the **full** path

Now enter a query in 'Master Query' that will extract the data you wish to use. For example

**Select SUPPLIER, SUPPLIERADDRESS1, SUPPLIERADDRESS2,
SUPPLIERADDRESS3, PRIMARYCONTACT from SUPPLIERMASTER**

Also enter a Supplier From and To parameter as follows

**From Supplier;SUPPLIERMASTER.SUPPLIER >= :FSUPPLIER
From Supplier;SUPPLIERMASTER.SUPPLIER <= :TSUPPLIER
Both linked to Code Type 1001**

'Save' the entries

Step 2. Run the Analysis and create the Data Source

This step will create the extracted data format from which the Merge Template can be created.

Click the '**Preview**' button to produce an Analysis View of the extracted data.
Click on '**Export**' on the top toolbar and select '**Mail Merge**'

This routine will generate your **SuppSource** data file containing the data displayed in the Analysis View.

The first time through you will also get a message stating '**File Not Found. Make sure the file path is correct**'. This is because the Word document to print the information has not been created yet. That is the next step.

Step 3. Create the Word Document and link to the Data Source

This step will create the Document using Microsoft Word and link to the generated fields in the extracted **SuppSource** file.

Go into Microsoft Word and create a new document and 'Save' it as **SuppLetter** in the path that you identified above.

Click on '**Tools**' on the top toolbar within Word and select '**Mail Merge**'. A panel will appear into which you should carry out the following steps.

Step 3.1. Click the '**Create**' button and select '**Form Letters**'

Step 3.2. On the displayed panel select 'Active Window'

Click on '**Get Data**' button and then on the **Open Data Source** option and point it the **SuppSource** document that you generated above. A panel will appear asking you to define what the Field and Record delimiters are. Select the following

Field Delimiter - , (I.e. A comma)

Record Delimiter - (enter)

Click the '**OK**' button

Step 3.3. On the presented panel click the '**Edit Main Document**' button.

You have now linked the document to the data source. The next step is to select the data fields and place them on the document.

Step 4. Constructing the Merge Document

This step will create the Document using Microsoft Word and link to the generated fields in

In the new document that you have just generated you will notice that a new toolbar called '**Mail Merge**' appears at the top of the screen. In that Toolbar click on the '**Insert Merge Field**' and you will see the field names in the **SuppSource** file.

Type in the word '**To**' followed by a space on the first line. The position of the cursor now defined where the first 'merged field' will appear.

From the '**Insert Merge Field**' select field '**SUPPLIER**'. Your document should now look like this

To << SUPPLIER >>

Complete the remainder of the document using a combination of typed words and linked Data Fields to produce the following

To <<SUPPLIER>>
<<SUPPLIERADDRESS1>>
<<SUPPLIERADDRESS2>>
<<SUPPLIERADDRESS3>>

Dear <<PRIMARYCONTACT>>

This is my test document for merging.

Regards

then 'Save' the document

Step 5. Preview

You can preview the finished report by clicking on the '<<ABC>>' button on the 'Mail Merge' Toolbar and, using the right and left arrows on the same Toolbar, progress through all the extracted records. You have now created the **SuppLetter** document to use in conjunction with the **SuppSource** file. Close all documents and we will now see the full process flow

Step 6. Running the Merged document from within Ostendo

Go into Ostendo and select *Purchasing>Views* then select '**Suppliers**'.

Click on '**Export**' on the top toolbar and select '**Mail Merge**' and you will be presented with the Merge document as you created above.

If you click on the print icon you will print the design view that you currently see
If you click on *File>Print* you will print the merged details
You can also click on the '**Merge**' button on the Merge Toolbar and print the merged details

15.3 Chart Views

The Chart View enables you to see pictorial representations of various statistics with option to drill down, filter, select chart format, etc.

In these exercises we will:

- Create a new Chart View of Items within Categories showing current stock against Re-Order Level
- Look at some of the actions that can be carried out within a Chart View

15.3.1 Creating a Chart View

Go into the Report and View Developer via *File>Reporting Configuration>Report and View Developer*.

You will see all the Reports, Views, Charts and Pivot Views that are supplied with Ostendo (These are held at the 'System' Level). Any new Report is automatically created with a Level of 'Company'

Click on the 'Add' button and, in the presented panel enter the following:

- 'Check' the '**Create from Scratch**' Radio Button
 - 'Check' the '**Chart**' Radio Button
-

- Name: enter '**Item Stock Levels**'
- Click the '**Create**' button

You will be taken to the '**Master Settings**' tab of the Report layout.

Step 1. Naming the Chart

In the upper part of the screen enter the following:

- **Include in Main Menu** is 'ticked' to denote that it will appear in the Ostendo menu
- **Name:** Leave as '**Item Stock Levels**'
- **Menu Order:** enter a number to denote where it appears in the '**Views**' Menu
- **Category:** From the drop-down list select '**Inventory**'
- **Title:** '**Item Stock Levels**'

Step 2. Defining the Data

The next step is to tell Ostendo from which table and what data is to be extracted. For our exercise we will extract all data from the Item Master table.

In defining the data for 'Chart' Views there are three distinct areas that must be specified in the Master Query.

The Fields that will facilitate the 'Drill-Down' options
 The Fields that are the displayed Values
 The lowest level that can be displayed

Click on the '**SQL Builder**' button to the right of the '**Master Query**' panel. In the presented screen locate '**ITEMMASTER**' in the '**Tables**' panel. (Hint: type I to take you straight to I in the list). 'Double Click' on this and you will see '**Select from ITEMMASTER**' appear in the '**Query**' panel with the cursor located after the word 'Select'.

Place the cursor between 'Select' and 'from' and add the following

Drill Down Fields: Use **ITEMCATEGORY, ITEMSTATUS**, (Note the commas)
Value Fields: Use **ONHANDQTY, REORDERLEVEL**, (Note: These must be numeric fields)
Lowest level Field: Use **ITEMCODE as ChartCategory**

The whole query will look like this

```
Select ITEMCATEGORY, ITEMSTATUS,
ONHANDQTY,
REORDERLEVEL,
ITEMCODE as ChartCategory
from ITEMMASTER
```

Click on the '**Preview**' button to the right of the screen to see the results

15.3.2 Viewing the Chart

If you exit the Report and View Developer and then go to **Inventory>Views** you will see your new Chart in the list. If you select the Chart you can manipulate the details of that screen as outlines in the following exercises

1. Main Toolbar

Data Levels: This shows the drill-down data levels available on each view. You can either select the Data Level by clicking on the Level Name. Alternatively you can 'left mouse' on the chart diagram to drill down one level at a time and 'right mouse' to go back up the levels

Selection within Level: If you click on the field to the right of each Level then a drop-down list is presented showing all the options within that level. Selecting from the drop-down list will restrict the display to that option

Customize Chart: Clicking on this button will allow you to customise the presentation. The presentation options are:

Series: Within this panel you can select:

- Select the field(s) to be displayed
- Drag and drop the field to the required appearance order

Data Groups: Not currently used

Options: Within this panel you can:

- Define the position of the Chart's Legend
- Define, within the above selection, where the Legend is to appear
- Define the orientation of the Legend
- Define if the Legend has a border around it
- Define if the Legend 'key' has a border around it
- Define the position of the Chart's Title
- Define, within the above selection, where the Title is to appear
- Define where the Toolbar is to appear
- Define if the Toolbar has a border
- Identify if the Diagram selector is to be visible
- Select if you want 'hints' to be visible when cursor is moved over the display

Column Diagram: From the drop-down list select the display format of the Chart. The options are:

- Column Diagram
- Bar Diagram
- Line Diagram
- Area Diagram
- Pie Diagram

2. Output Options

Print: You can output the current view to your screen prior to printing or print directly to a selected printer.

Email: The drop-down list provides the following options

XLS - Generates an XLS (spreadsheet) from the displayed information and then opens up the Emailing routine on your PC with this XLS document as an attachment

HTML - Generates an HTML format and displays this in the 'body' of the generated email

XML - Generates an XML document, which can be called and displayed as a standard Web form. It also creates an XSL document showing the source document from the displayed information. The program then opens up the Emailing routine on your PC with the XML and XSL documents attached.

CSV - Generates an CSV (Comma Separated Value) document from the displayed information and then opens up the Emailing routine on your PC with this CSV document as an attachment

Export: The drop-down list provides the following options

XLS - If this is selected then a panel will appear for you to save the current displayed data

as an Excel Spreadsheet. You may then call up Excel and manipulate the data as required.

HTML - If this is selected then a panel will appear for you to save the current displayed data as an HTML file. This file can then be called and displayed as a standard Web form using your Browser.

XML - If this is selected then a panel will appear for you to save the current displayed data as an XML file. Two files are created.

- An XML document that can be called and displayed as a standard Web form using your Browser.
- An XSL document showing the source document
This file can then be called and displayed as a standard Web form using your Browser.

CSV - If this is selected then a panel will appear for you to save the current displayed data as a Comma Separated Value file. This can be used as an import format to external systems. You may also call up the CSV file in Excel (File Type = .csv) and manipulate the data as required.

15.4 Pivot Views

When looking for the tools of analysing data in different ways without any coding, the pivot grid is the best solution. With the drag-and-drop functionality, it allows the User to rearrange fields in the view dynamically. This includes:

- Interchanging columns and rows on the fly
- Filtering and sorting items in different ways
- Collapsing and expanding data at different levels.

In this exercise we will create Pivot View of Inventory Value

15.4.1 Creating a Pivot View

Go into the Report and View Developer via **File>Reporting Configuration>Report and View Developer**.

Click on the 'Add' button and, in the presented panel enter the following:

- 'Check' the '**Create from Scratch**' Radio Button
- 'Check' the '**Pivot**' Radio Button
- Name: enter '**Inventory Value**'
- Click the '**Create**' button

You will be taken to the '**Master Settings**' tab of the Report layout.

Step 1. Naming the Pivot View

In the upper part of the screen enter the following:

- **Include in Main Menu** is 'ticked' to denote that it will appear in the Ostendo menu
- **Name**: Leave as '**Inventory Value**'
- **Menu Order**: enter a number to denote where it appears in the '**Views**' Menu
- **Category**: From the drop-down list select '**Inventory**'
- **Title**: '**Inventory**'

Step 2. Defining the Data

The next step is to tell Ostendo from which table and what data is to be extracted. For our

exercise we will extract data from 3 different Tables

- Item Master
- Warehouse
- Units Of Measure

In defining the data for 'Pivot' Views there are two distinct areas that must be specified in the Master Query.

- The Fields that will facilitate the 'data selection' options
- The Fields that are the displayed Values

As the query comes from 3 different tables Ostendo will also require you to:

- Identify a key field that provides a Unique Record ID.

The above two areas plus the unique ID are all contained in the following Query

```
select inventory.warehousecode, inventory.locationcode, itemmaster.itemcategory,  
itemmaster.itemcode,  
cast((itemmaster.averagecost * (inventory.inventoryqty * itemunits.conversionfactor)) as  
decimal(13,2)) as "Value at Average",  
cast((itemmaster.standardcost * (inventory.inventoryqty * itemunits.conversionfactor)) as  
decimal(13,2)) as "Value at Standard",  
inventory.sysuniqueid as griduniqueindex  
from Inventory,Itemmaster,itemunits where  
Itemmaster.itemcode = inventory.itemcode and Itemunits.itemcode = inventory.itemcode  
and itemunits.tounit = inventory.inventoryunit
```

Let's look at the query in more detail

- The first line simply selects fields from the Inventory and Item Master tables
- The two 'Cast' statements calculate the Value based on the Cost * Stock Quantity * UOM Conversion factor
- The next line is the Unique ID field associated with a pre-defined 'griduniqueindex'
- The remainder links the tables together

Click on the **Preview** button to the right of the screen to see the results

15.4.2 Viewing the Pivot View

If you exit the Report and View Developer and then go to **Inventory>Views** you will see your new Pivot View in the list. If you select the Pivot you can manipulate the details of that screen as outlines in the following exercises

1. Basic View

A Pivot Grid in Ostendo has four major 'working' areas shown in the top of the display

Filter Band: A band across the top of the Grid where the selected fields from the Query are displayed. If you click on the drop-down contained within each field then you can specify the specific selection(s) to appear in the Pivot Grid.

Try doing the following

- Click on 'Item Category' and drag and drop over the words 'Drop Row Fields Here'
- Click on 'Item Code' and drag it to the right of 'Item Category' and drop it
- Click on 'Warehouse Code' and drag and drop over the words 'Drop Column Fields Here'
- Click on 'Location Code' and drag it to the right of 'Warehouse Code' and drop it
- Click on 'Value at Standard' and drag and drop over the words 'Drop Data Fields Here'
- Click on 'Value at Average' and drag it to the right of 'Value at Standard' and drop it

2. Field Selection

You can carry out the following activities that enable you to address the visibility of selected fields.

Individual Field visibility options: If you 'right mouse on any field in the upper 'work area' then the following options are available

- **Hide:** Hides the selected field from the display
- **Order:** Moves the selected field to:
 - Move to Beginning
 - Move to End
 - Move Right
 - Move Left
- **Hide or Show Field List:** Hides or shows a panel in the bottom right of the screen for hiding and restoring multiple fields. See next option

Multiple Field visibility options: If you 'right mouse outside of any field in the upper 'work area' then the following is displayed

- **Hide or Show Field List:** Hides or shows a panel in the bottom right of the screen for hiding and restoring multiple fields. Within this panel you can
 - Drag any shown field onto the Pivot Grid for use in the Grid
 - Drag any field from the Pivot Grid to this panel to hide the field

3. Expanding, Collapsing, Grouping of Columns and Rows

You can click on the 'Plus' or 'Minus' buttons shown in the heading of fields to expand and collapse columns and rows that have nested columns and rows:

4. Filtering

Data can be filtered against column, row and filter fields by clicking the filter button (Black downward facing triangle seen in the field header) and selecting items in the filter drop-down.

5. Field Drill-Down

For any calculated field in the main body of the Pivot Grid you can 'double click' on the field to drill down and display the records that are the source of the displayed summary information.

6. Keyboard Shortcuts

The keyboard shortcuts and mouse operations listed in the table below can be used by the User to move focus between cells.

- Clicking a cell: Focuses the clicked cell.

- Pressing the **Up, Down, Left or Right arrow** keys: Focuses a corresponding adjacent cell.
- Pressing the **Page Down** key: Moves row focus one page down preserving the column focus.
- Pressing the **Page Up** key: Moves row focus one page up preserving the column focus.
- Pressing the **Home** key: Focuses the first cell within the current row.
- Pressing the **End** key: Focuses the last cell within the current row.
- Pressing the **Ctrl+Home** key: Focuses the first cell within the first row.
- Pressing the **Ctrl+End** key: Focuses the last cell within the last row.

7. Re-Ordering Fields

The following operations can be performed by Users to rearrange fields:

- Drag a field header and drop it at a new position within the same or another header area.
- Select an option from the "Order" submenu in the field header context menu.
- Double-click a field header within the customisation form or selecting a field header and clicking the form's **Add To** button.

8. Resizing Columns

The following operations can be performed by Users to resize columns:

- Drag the right edge of a column header to change the width of the current and relative columns.
- If you Double-click the right edge of a column header then it applies best fit to the current and relative columns

9. Selecting Cells

The Pivot Grid control allows Users to select multiple data cells using the mouse or the available keyboard shortcuts. The data that is displayed by the selected cells can be copied to the clipboard and pasted into other applications (e.g. MS Excel, MS Word).

Pressing the **Shift+Arrow key** combination enables Users to select a continuous range of data cells. When such actions are carried out any previous cell selection is cleared.

Pressing the **Ctrl+A** key combination will select all data cells within the data area.

10. Copying Selected Records to the Clipboard

Pressing the **Ctrl+C** or **Ctrl+Ins** key combination will copy the selected data cells to the clipboard as text.

11. Sorting

The values of column fields and row fields are always displayed in ascending or descending order. You can alternate between ascending and descending by simply clicking on the field header.

12. Output Options

Print: You can output the current view to your screen prior to printing or print directly to a selected printer.

Email: The drop-down list provides the following options

XLS - Generates an XLS (spreadsheet) from the displayed information and then opens up the Emailing routine on your PC with this XLS document as an attachment

HTML - Generates an HTML format and displays this in the 'body' of the generated email

XML - Generates an XML document, which can be called and displayed as a standard Web form. It also creates an XSL document showing the source document from the displayed information. The program then opens up the Emailing routine on your PC with the XML and XSL documents attached.

CSV - Generates an CSV (Comma Separated Value) document from the displayed information and then opens up the Emailing routine on your PC with this CSV document as an attachment

Export: The drop-down list provides the following options

XLS - If this is selected then a panel will appear for you to save the current displayed data as an Excel Spreadsheet. You may then call up Excel and manipulate the data as required.

HTML - If this is selected then a panel will appear for you to save the current displayed data as an HTML file. This file can then be called and displayed as a standard Web form using your Browser.

XML - If this is selected then a panel will appear for you to save the current displayed data as an XML file. Two files are created.

- An XML document that can be called and displayed as a standard Web form using your Browser.
- An XSL document showing the source document
This file can then be called and displayed as a standard Web form using your Browser.

CSV - If this is selected then a panel will appear for you to save the current displayed data as a Comma Separated Value file. This can be used as an import format to external systems. You may also call up the CSV file in Excel (File Type = .csv) and manipulate the data as required.

15.5 Inquiry Screens

'Inquiry Screens' in Ostendo provide facility for Users to create their own Inquiry Screens across all sections of Ostendo and add that Inquiry Screen to the main Ostendo Menu

In these exercises we will:

- Create a new Inquiry with data from the Item Master table
- Look at some of the actions that can be carried out with an Inquiry Screen
- Add a sub detail Inquiry Screen showing the Item's alternate Units of Measure

15.5.1 Creating an Inquiry Screen

Go into the Report and View Developer via **File>Reporting Configuration>Report and View Developer**.

You will see all the Reports, Inquiry Screens, Views, Charts and Pivot Views that are supplied with Ostendo (These are held at the 'System' Level). Any new Report/View is automatically created with a Level of 'Company'

Click on the 'Add' button and, in the presented panel enter the following:

- 'Check' the '**Create from Scratch**' Radio Button

- 'Check' the '**Inquiry**' Radio Button
- Name: enter '**Item Inquiry**'
- Click the '**Create**' button

You will be taken to the '**Master Settings**' tab of the Report layout.

Step 1. Naming the View

In the upper part of the screen enter the following:

- **Include in Main Menu** is 'ticked' to denote that it will appear in the Ostendo menu
- **Name:** Leave as '**Item Inquiry**'
- **Menu Order:** enter a number to denote where it appears in the '**Inquiry**' Menu
- **Category:** From the drop-down list select '**Inventory**'
- **Title:** '**Item Inquiry**'

Step 2. Defining the Data

The next step is to tell Ostendo from which table and what data is to be extracted. You should note that, for Inquiry Screens, it is recommended that you define the specific fields to be extracted. If you define (for example) **Select * from.....** then a delay will occur when generating the screen during daily use

For our exercise we will extract selected data from the Item Master table. You can click on the '**SQL Builder**' button to the right of the '**Master Query**' panel and create your own query or simply copy and paste this query

Select ITEMCODE as "Item_Code", ITEMDESCRIPTION as "Description", ITEMUNIT as "Item_Unit",SYSUNIQUEID from ITEMMASTER

Points to note about this query

You can redefine what will be shown in the column title as follows:

- immediately follow the selected field with **as**
- The redefined title should be enclosed in quotation marks " "
- Words within redefined titles should be separated with an underscore _
- The Underscore will be removed when the Inquiry screen is displayed
- The Master query selection **MUST** contain field of **SYSUNIQUEID**
- Colour filled fields in the Inquiry

The Tracking Codes defined under Assembly, Purchase, Sales and Job settings allow you to associate a colour code with the Tracking status. You can display this colour in an Inquiry View by using the following format

"DISPLAYCOLOUR_TrackingColour"

Step 3. Test the Inquiry Screen

Click on the '**Preview**' button to the right of the screen to see the results

Viewing the new Inquiry Screen

If you exit the Report and View Developer and then go to **Inventory>Inquiry** you will see your new Inquiry Screen in the list. Select the Inquiry Screen. You can manipulate this screen as outlined in the following exercises

15.5.2 Working the Inquiry Screen

The main Inquiry 'List' screen shows the retrieved information from your database. You can select a line in this view and click on the 'Detail' tab to see the record's details in a 'Card' View

Back to the 'List' screen. The extracted records can be 'sliced and diced' in the following manner.

Search & Sort

The following features are available that allow you to sort and filter the displayed data.

- sort any column in the displayed records into ascending or descending sequence by clicking on the selected column heading
- you may 'filter' the records based upon your own selection criteria by clicking on the blue triangular symbol in the selected column heading. From the drop-down list you may select the following:-
 - All - displays all records
 - Blanks - Displays only those records that have no data in the field
 - Non-Blanks - Displays only those records that contain data in this field
 - Select the specific field content by using the 'Custom' option

Moving and Hiding Columns

You may also move columns as well as take unwanted columns from the display

- You can move columns by clicking on the column heading and 'dragging' the column into the required position.
- If you double click on the Column Heading's right edge you can make the column automatically 'close up' to match the amount of data in the field. You can also achieve this by 'Right Clicking' the column heading and selecting 'Best Fit'.
- If you wish to 'Close Up' all the columns then you should 'Right Click' on any column heading and select 'Best Fit - All Columns'.
- If you 'Right Click' on a column Heading and select 'Field Chooser' then a panel will appear for storing unwanted columns and enables you to 'Customise' the screen. You can move any unwanted columns to this panel by simply clicking on the column heading and dragging the column into this panel. You can recall stored columns by reversing this procedure.
An alternative to this is to 'Right Click' on a column Heading and select 'Remove this column'. This will automatically place the unwanted column in the storage panel
To hide the 'Field Chooser' Panel simply click on the 'x' in the upper right corner of the panel

Grouping

You may wish to Group 'like' records. To do this simply drag the required column heading into the area at the top of the screen where it states 'Drag a column header here to group by that column'. (If this area is not visible then you should 'Right Click' on any column heading and select 'Group By Box')

The screen will now group all records where the content of the 'Grouped' field is the same.

- If you click on the '+' indicator against each Group you can see the detailed records.
- This Grouping facility is not just single level. You can Group within Group, etc by simply dragging and dropping the 'sub-group' to the right of the first Group.
- This can be repeated for as many levels as you require.
- An alternative method of achieving this is to 'Right Click' on the selected column heading

and select 'Group By This Field')

Summary Totals

For each Group you may wish to display summary totals. To do this, expand a Group by clicking on the '+' indicator to display the individual records. At the bottom of the Group List is a blank area. If this area is not visible then 'Right Click' on a column heading and select 'Group Footers'.

Now go to this blank area under any column and 'right click' the mouse to display the following options. These can be accessed depending upon the type of field (For example, you cannot 'Sum' a Date field)

- | | |
|-----------|------------------------------|
| ▪ Sum | Numeric fields only |
| ▪ Min | Numeric and Date fields only |
| ▪ Max | Numeric and Date fields only |
| ▪ Count | All fields |
| ▪ Average | Numeric fields only |
| ▪ None | All fields |

The selected column will now display the summary information

Grand Totals

You may also wish to display grand totals for the displayed data. To do this you should see a blank area at the end of the displayed list. If this area is not visible then 'Right Click' on any column Heading and select 'Footer'.

As with Group Footers, you can go to the blank area under any column and 'right click' the mouse to display the options.

15.5.3 Adding a 'drill-down' Inquiry

Go back to the Report and View Developer via **File>Reporting Configuration>Report and View Developer** and recall the Item Inquiry Screen. We will now extend the Inquiry from a simple Item Inquiry to looking at all Item Units linked to each Item.

Step 1. Creating the Detail Query

Click on the **'Detail Query'** tab and add this query

```
select ITEMCODE as "Item_Code", TOUNIT as "To_Unit", CONVERSIONFACTOR as
"Conv_Factor",DISCOUNTPERCENT as "Discount", SYSUNIQUEID from ITEMUNITS where
ITEMCODE = :Item_Code
```

You will note that the link to the main query is to the renamed field . I.e. Item_Code and not ITEMCODE.

Click on the **'Save'** button to save the query

Step 2. Test the Inquiry Screen

Go back to the **'Master Settings'** tab and click on the **'Preview'** button to the right of the screen. You will see that an extra (**Lines**) tab is now available for you to drill down and view the Units of Measure held against the selected Item.

Step 3. Working the Inquiry

The same options in working the Main Inquiry List screen also applies to this 'Lines' screen

16 15. Beginners Guide to Scripting

These Exercises are designed to provide an introduction to Scripting in Ostendo and give you an insight into the potential that can be achieved through using this feature.

The Tutorial will go through the following steps

- Your first script
- Basic Structure of a Script
- Standard routines and their Syntax
- Pre-defined Ostendo Functions and Procedures
- Where Scripts are used in Ostendo

Throughout this Tutorial you will find ready-made examples. To see how these work simply go into **File>Custom Scripts** in Ostendo and add a new script. Click on the **'Script'** tab then 'Copy and Paste' the example.

You can then **'Save'** the Script and click on the **'Test Script'** Button to see how it works

16.1 The First Script

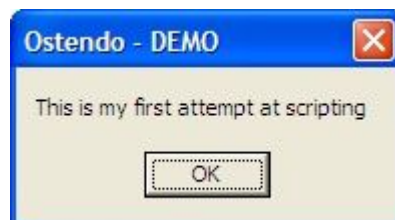
Here is a simple script that show you how scripting works

Go into **File>Custom Scripts** and add a new script called (say) **MyMessage**

Click on the **'Script'** tab and enter the following Script (You can copy and paste this if required)

```
// My First Script
Begin
  showmessage('This is my first attempt at scripting');
End.
```

Click on the **'Save'** Button and then click the **'Test Script'** button. You should now see the following panel



Now that you see how it works let's take it a step further

16.2 Basic Structure

This section describes the basic elements to a Script

16.2.1 Introduction

Script structure is made of two major blocks:

- Variables and Constants declarations
- Main block.

Both are optional, but at least one should be present in the script.

For example:

```
//This defines the Variable
var
  MyVariable: string;
//This is the Main Block
begin
  MyVariable:= 'This is my second attempt at scripting';
  showmessage(MyVariable);
end.
```

Let's look at the above coding in a little more detail

In the above example you should note the following:-

- Lines starting with `//` are comment lines and will be ignored when the Script is run..
- Each line of distinct instruction ends in a semi-colon (`;`)
- A group (or block) of instructions to carry out a task should have a `begin` and `end` command.
- It is good practice to offset sub-lines within a program
- We have defined a variable '`MyVariable`' (a 'string' means that it is non-numeric).
- The (`MyVariable`) after `showmessage` tells the program to print the content of the variable string.
- The whole Script is concluded with a 'Full Stop' after the final '`End`'

16.2.2 Identifiers

Identifier names in script (Variable and Constant Names, Function and Procedure names, etc.) should begin with a character (a..z or A..Z), or '`_`', and can be followed by alphanumeric chars or '`_`' char. They cannot contain any other character or spaces.

Valid Examples are :

```
VarName
_Some
V1A2
_____Some_____
```

None-Valid Examples are :

```
2Var
My Name
Some-more
This,is,not,valid
```

16.2.3 Character 'String'

Strings (I.e. A sequence of characters) are declared using single quote (') character. Double quotes (") are not used. Be aware that if you are using a Word Processor to create the script it may automatically convert the ' into ì (Speech Mark) and that is not a valid character. A String can include Variables and Constants but these must be segregated by use of a + sign. For example

```
A:='This is a text';  
B:='String with ' + Var1 + ' in the middle';
```

16.2.4 Comments

Comments can be inserted inside script. You can use // chars or (* *) or { } blocks. Using // char the comment will finish at the end of line.

```
Begin  
// This is a comment before ShowMessage  
ShowMessage('Ok');  
(* This is another comment *)  
ShowMessage('More ok!');  
{ And this is a comment  
covering multiple lines }  
ShowMessage('End of okays!');  
End.
```

16.2.5 Variables and Constants

A **Variable** is where variable data can be stored. You must declare a Variable. The Variable Type is not mandatory but it is recommended that you declare it. For Example:

```
var  
S;  
A: String;
```

Common Variable Types are:

```
String - AlphaNumeric characters  
Double - Number with decimals  
Integer - Number with no decimals  
Boolean - True or False
```

There are a few more but we will not be addressing these in this Introductory Guide

A **Constant** is - effectively - a 'fixed' Variable and is defined in a similar way to Variables. For example:

```
Const  
Pi = 3.14159;  
E = 2.71828496;  
MyConstant = 1760;
```

16.3 Standard Routines

The following standard routines are used in scripting. These contain 'Reserved Words' and tell the program to carry out specific tasks.

- **begin .. end** constructor
- **procedure** and **function** declarations
- **if .. then .. else** constructor
- **for .. to .. do** constructor
- **arrays**
- **case** statements

16.3.1 Begin - End Constructor

The **begin** and **end** constructors define a 'block' of related instructions to perform a specific action. These blocks can be 'nested' (i.e. An instruction Block could contain sub-level instructions, each which is contained in a Begin...End Block. You will see how these work in the following examples.

16.3.2 Procedure and Function Declaration

There is no distinct difference between a Procedure and a Function although they could be categorised as

- A **Function** defines a subroutine that returns a value, whereas
- A **Procedure** defines a subroutine that does not return a value

An example of a **Procedure** is where you run a query to update a record

```
Procedure HelloWorld;  
  begin  
    ShowMessage('Hello world!');  
  end;  
Begin  
  HelloWorld;  
End.
```

An example of a **Function** is

```
Var  
  TheValue: Integer;  
Function GetValue: Integer;  
Begin  
  result := AskQuestionNumericRange('Please enter a number between 1 and 9','INTEGER',  
  1,9,1);  
end;  
// The 'result' of the function  
Begin  
  TheValue := GetValue;  
  Showmessage(TheValue);  
End.
```

16.3.3 If - Then - Else

The **If...Then...Else** combination enables you to ask **IF** a certain condition exists and - if so - **THEN** carry out an Action - **ELSE** - if not - carry out some other action. The **ELSE** element is not mandatory. I.e. If it is not present then no 'Else' action will take place. Here are three examples of how the **If...Then...Else** routine works using Variables **J** and **Result**

Example 1

```
Var
J: Integer;
Result: String;
Begin
  J := AskQuestionNumericRange('Please enter a number between 1 and 9','INTEGER',"
,1,9,1);
  if J <> 2 then Result := 'True';
  Showmessage (Result);
End.
```

Example 2

```
Var
J: Integer;
Result: String;
Begin
  J := AskQuestionNumericRange('Please enter a number between 1 and 9','INTEGER',"
,1,9,1);
  if J = 2 then Result := 'False' else Result := 'True';
  Showmessage (Result);
End.
```

Example 3

```
Var
J: Integer;
Result: String;
Begin
  J := AskQuestionNumericRange('Please enter a number between 1 and 9','INTEGER',"
,1,9,1);
  if J <> 2 then
    begin
      Result := 'True';
    end
  else
    Result := 'False';
  Showmessage (Result);
End.
```

16.3.4 For - To - Do

The **For...To...Do** combination enables you to carry out a Block of Script then loop back and process it against. This can be repeated for the number of time you specify.

This example demonstrates how this works. It uses a procedure that firstly identifies a start number and then repeats itself whilst at the same time incrementing the number.

```

// Define the Variable
var
  x: integer;
// Define the Procedure
Procedure Looping;
Begin
  for x := 1 to 8 do
    Showmessage('The Current Count is ' + IntToStr(x));
  End;
// Run the Procedure
Begin
  Looping
End.

```

You should note the following

- A variable (**x**) has been declared as an integer (whole number, no decimals)
- The **'for'** statement defines **x** as being 1 through to 8
- The **'do'** statement tells the program to repeat for each incremental whole number
- Showmessages can only contain Text therefore the Integer **'x'** is converted to a String using the IntToStr command
- The above gives an example of creating a Procedure in the first 'Block' and then executing the procedure in the final 'Block'.

Run the process and view the result.

You will see that whenever you click on any displayed button the **'x'** will increment and you will see the next number in the sequence. After number 8 the whole procedure will be completed and the panel will disappear.

16.3.5 Arrays

Scripting supports arrays which are multiple variants from which a single or multiple variants can be selected. You can then access an array using an index. You should note that arrays in script always use 0 as the start index. Here is an example of an Array

```

Var
  TheArray: Array of String;
Begin
  TheArray := ['green','red','blue'];
  Showmessage(TheArray[1]);
end.

```

This will return **'red'** because that is the second option in the Array

16.3.6 Case Statements

A Case Statement is where you declare possible conditions and, via a scripting block, define what to do when that condition is encountered. You can optionally have an **'else'** section to the Case Statement

Example:

```

Var
  CaseNumber: Integer;
Begin

```

```

CaseNumber:= AskQuestionNumericRange('Please Type in your option number','INTEGER',
'The normal options are 1 to 3',1,4,1);
case (CaseNumber) of
'1': ShowMessage('green');
'2': ShowMessage('orange');
'3': ShowMessage('red');
else
  ShowMessage('black');
End;
End.

```

16.4 Pre-defined Ostendo Functions

There are a number of Pre-Defined Functions and Procedures created specifically for running with Ostendo. These allow you to provide 'High-Level' data and the Function or Procedure will do the rest. The individual Functions can be combined to provide a complete Script covering (1) What data is required to commence the process, and (2) What to do with that data.

These exercises will not cover all the Pre-Defined Functions but will provide you with sufficient knowledge to get you started.

Let us begin by looking at functions that get data from various sources. This data is then used further within the script. These Functions collect:

- User entered Data
- Data from Ostendo Database
- Data from external sources

16.4.1 User Entered Data

The following functions allow you to enter, or select, data to be used within the script.

```

AskQuestion
AskQuestionWithLookup
AskMandatoryQuestion
AskMandatoryQuestionWithLookup
AskQuestionNumericRange
AskQuestionWithUserDefinedLookup
DisplayData

```

16.4.1.1 AskQuestion

When run, this Function displays a panel into which the User enters some data. This data is stored against a Variable for subsequent use in the Script. This type of question can optionally include a pre-defined list from which the answer is selected. The elements that make up this function are:

Variable: The defined variable against which the result will be held.

Question: Define the question (max 200chars) enclosed on 'commas'

QuestionType: This is INTEGER, TEXT, NUMERIC, DATE, BOOLEAN, TEXT LIST

FullExplanation: A longer explanation that can be referenced during configuration

[d]: Optional entry to define a default display entry from the List

ValueList: Allowable options - separated by a comma

Default Value: Optional Entry to prefill question with current answer if re-doing questions

The following example defines a Variable '**OptionSelect**' and then asks the question from which you can (optionally) type in an answer. After keying in the option a message will be presented

showing the value now held in variable OptionSelect

```
// Define the Variable
Var
  OptionSelect: String;
// You can then ask the question to answer OptionSelect using
Begin
  OptionSelect := AskQuestion('Please Type in your option','TEXT','You can leave blank if
you wish,');
  Showmessage('Your text entry is ' + OptionSelect);
End.
```

To see this in action go to **File>Custom Scripts** and create a new script called (say) '**Question**'. Click on the '**Script**' tab then copy and paste the above script and '**Save**' the entry. If you click on the '**Test Script**' button then the Script will be run.

Try adding a '**Value List**' (Example: **Blue,Green,Red**). Insert this between the final quotes in the AskQuestion function. For example

```
OptionSelect := AskQuestion('Please Type in your option','TEXT','You can leave blank if
you wish','Blue,Green,Red');
```

Run the script and see the result.

Finally, amend the '**Value List**' to read **Blue,[d]Green,Red** and you will see that the colour **Green** will prefill the entry field instead of the first in the list

16.4.1.2 AskQuestionWithLookup

This is similar to Ask Question but, in this instance you specify an Ostendo Table from which to make a selection. A list of Table Numbers can be seen under **Help>Scripting** and selecting '**Lookup Numbers**'. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
Question: Define the question (max 200chars) enclosed on 'commas'
Full Explanation: A longer explanation that can be referenced during configuration
LookupIndex: The Index Reference of the Ostendo Table
Default Value: Optional Entry to prefill question with current answer if re-doing questions

In the following exercise we will get the Function to look at the Item Maser Table from which a selection can be made.

```
// Define the Variable
Var
  SelectedItem: String;
// You can then ask the question to answer SelectedItem using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code',
1004,);
  Showmessage('Your selection is ' + SelectedItem);
End.
```

16.4.1.3 AskMandatoryQuestion

This is similar to function AskQuestion. However, in this instance you MUST answer the question before you can continue. Try this script

```
// Define the Variable
Var
  TimberType: String;
// You can then ask the question to answer TimberType using
Begin
  TimberType := AskMandatoryQuestion('Please enter the timber type you require ', 'TEXT', '');
  Showmessage('Your selection is ' + TimberType);
End.
```

When the question is presented click on the 'Answer' button without making an entry and see what happens

16.4.1.4 AskMandatoryQuestionWithLookup

This is similar to function AskQuestionWithLookup. However, in this instance you MUST answer the question before you can continue. Try this script

```
// Define the Variable
Var
  SelectedItem: String;
// You can then ask the question to answer SelectedItem using
Begin
  SelectedItem := AskMandatoryQuestionWithLookup('Item Code', 'Please select the Item Code', 1004,);
  Showmessage('Your selection is ' + SelectedItem);
End.
```

When the question is presented click on the 'Answer' button without making an entry and see what happens.

Create your own Script to access the 'Customer Asset' table and show your selection via a 'Showmessage'

16.4.1.5 AskQuestionNumericRange

This question asks you to enter a numeric value. To support this you can optionally include a minimum and maximum value along with incremental steps. The elements of this Function are:

Variable: The defined variable against which the result will be held.

Question: Define the question (max 200 chars) enclosed on 'commas'

Question Type: this is either NUMERIC or INTEGER

Full Explanation: A longer explanation that can be referenced during configuration

MinValue: The numeric value denoting the minimum value allowed during data entry

MaxValue: The numeric value denoting the maximum value allowed during data entry

NumberInc: A numeric value denoting the incremental steps within the range. If nothing is entered then any value in the range will be accepted.

Default Value: Optional Entry to prefill question with current answer if re-doing questions

To see how this is used we will define a Variable 'DeskLength' and then ask the question for the user to enter the Length. After keying in the Length - which in our example must be a multiple of 10 - a message will be presented showing the value now held in variable DeskLength

```

// Define the Variable
Var
  DeskLength: Integer;
// You can then ask the question to answer DeskLength using
Begin
  DeskLength := AskQuestionNumericRange('Please enter the Length (mm) of the Desk',
'INTEGER','We can only make desks between 1000mm - 3000mm in length'
,1000,3000,10,DeskLength);
  Showmessage('Your Desk Length is ' + intostr(DeskLength));
// Note how we converted from Integer to String to complete the message
End.

```

16.4.1.6 AskQuestionWithUserDefinedLookup

This Function provides facility for you to define a Query to extract select records. From those records you can then select a specific Value. The elements that make up this function are:

LookupSQL: A standard Query enclosed in single quotes
Question: Define the question (max 200chars) enclosed in single Quotes
Full Explanation: A longer explanation that can be referenced during configuration
Default Value: Optional Entry to prefill question
LookupTitle: The title that will appear in the displayed panel
Result Field: The field from a selected record that returns the value
LookupHeight: Height of the displayed panel in pixels. Default = 320
LookupWidth: Width of the displayed panel in pixels. Default = 440

The following example defines a Variable '**TheLostQuote**' and then asks the question. The drop-down list shows data from the Query (I.e. All 'Lost Quotes'). After highlighting the selected line and clicking the '**OK**' button the 'Lost Quote' will - using the ExecuteSQL Function described later - have its status changed to '**Quote**' and a message will be presented showing that the action has been carried out

```

// Define the Variable
Var
  TheLostQuote: String;
// You can then select the data and ask the question for answering 'TheLostQuote'
Begin
  TheLostQuote := AskQuestionWithUserDefinedLookup('Select * from SalesHeader
where OrderStatus = "Lost"', 'Select the Lost Sales Quote', "", 'Select Quote', 'OrderNumber'
,350,500);
  Executesql('update SalesHeader set OrderStatus = "Quote" where OrderNumber = "' +
TheLostQuote + '"');
  Showmessage('Status of Lost Quote ' + TheLostQuote + ' has been changed to Quote');
End.

```

16.4.1.7 DisplayData

This Function allows you to create an inquiry screen to list Ostendo records extracted via an SQL. From the resultant list (which has additional search facilities included) you can select a specific record and return a field value from that record for further action. The elements that make up this function are:

SQL: A standard Query enclosed in single quotes
Title: The title that will appear in the displayed panel
Result Field: The field from a selected record that returns the value
FormHeight: Height of the displayed panel in pixels

FormWidth: Width of the displayed panel in pixels

In this exercise we will extract all Labour Codes. In the displayed panel we will select a specific Labour Code which will be held against Variable 'SelectedLabourCode'. In this example we have included an additional line to run another function that will update the Charge Rate of that Labour Code.

```
// Define the Variable to be populated by the selected Labour Code
Var
SelectedLabourCode: String;
// You can then ask the question to answer 'SelectedLabourCode' using
Begin
    SelectedLabourCode := DisplayData('Select * from LabourMaster', 'Labour Codes',
'LabourCode', 500, 1200);
    ExecuteSQL('update LabourMaster set StdSellRate = 55 where LabourCode = "' +
SelectedLabourCode + '"');
    Showmessage('Labour Code ' + SelectedLabourCode + ' updated!');
End.
```

16.4.2 Data from Ostendo Database

The following functions get data from the Ostendo database without User intervention. The data is stored against a Variable for subsequent use within the Script.

```
GetBooleanFromTable
GetStringFromTable
GetIntegerFromTable
GetDateFromTable
GetDoubleFromTable
GetCost
GetStdBuyPrice
GetStdSellPrice
GetSQLResult
```

16.4.2.1 GetBooleanFromTable

This will access the Ostendo Database and get the Boolean Value (True/False) of a field in a nominated record. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
TableName: The name of the Ostendo Table
FieldName: The Field Name within the Table containing the Boolean Value
KeyField: The Field against which you are identifying the specific record
KeyValue: The Specific record identity within the Key Field

In this exercise we will check a specific Item in the Item Master table to see if a Warranty record is to be generated when it is sold. We will use the AskQuestionWithLookup function to get the Item Code.

```
// Define the Variable to be populated by the result
Var
SelectedItem: String;
PartYesNo1: Boolean;
// You then construct the enquiry to answer 'PartYesNo1'
Begin
    SelectedItem := AskQuestionWithLookup('Item Code', 'Please select the Item Code',
```

```

1004,);
PartYesNo1 := GetBooleanFromTable ('ItemMaster','WarrantyApplies','ItemCode'
,SelectedItem);
Showmessage(PartYesNo1);
End.

```

Try amending the above script to access the Descriptor Table and carry out the same check

16.4.2.2 GetStringFromTable

This will access the Ostendo Database and get the contents of a field in a nominated record. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
TableName: The name of the Ostendo Table containing the string
FieldName: The Field Name within the Table containing the string
KeyField: The Field against which you are identifying the specific record
KeyValue: The Specific record identity within the Key Field

In this example we will access the Item Master and, for a selected Item, extract the Primary Supplier. The script begins with AskQuestionWithLookup and upon the Item being selected will return a message showing the Primary Supplier. In this script we have added an 'If..Then' condition to cover where the Item does not have a Primary Supplier

```

// Define the Variable
Var
  SelectedItem: String;
  PrimSupp: String;
// You can then get the PrimarySupplier using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  PrimSupp := GetStringFromTable('ItemMaster','PrimarySupplier','ItemCode'
,SelectedItem);
  If PrimSupp = " then
    Begin
      Showmessage('Item ' + SelectedItem + ' has no Primary Supplier');
    end
  else
    Showmessage('Primary Supplier for Item ' + SelectedItem + ' is ' + PrimSupp);
  End.

```

16.4.2.3 GetIntegerFromTable

This will access the Ostendo Database and get the Integer Value of a field in a nominated record. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
TableName: The name of the Ostendo Table
FieldName: The Field Name within the Table containing the number
KeyField: The Field against which you are identifying the specific record
KeyValue: The Specific record identity within the Key Field

In this example we will access the Item Master and, for a selected Item, extract the Leadtime. The script begins with AskQuestionWithLookup and upon the Item being selected will return a message showing the Leadtime.

```

// Define the Variable
Var
  SelectedItem: String;
  TheLeadTime: Integer;
// You can then get the Leadtime using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  TheLeadTime := GetIntegerFromTable('ItemMaster','LeadTime','Itemcode'
,SelectedItem);
  Showmessage('The Leadtime is ' + inttostr(TheLeadTime) + ' days');
End.

```

Note how we converted TheLeadTime to a string value. This is because a Showmessage will only accept String Values

16.4.2.4 GetDateFromTable

This will access the Ostendo Database and get the date currently held in a field in a nominated record. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
TableName: The name of the Ostendo Table
FieldName: The Field Name within the Table containing the date
KeyField: The Field against which you are identifying the specific record
KeyValue: The Specific record identity within the Key Field

In this exercise we will access the Ostendo Database and get the '**Last Cost Date**' from a selected Descriptor record

```

// Define the Variable
Var
  SelectedDescriptor: String;
  TheLastCostDate: TDate;
// You can then get the date using
Begin
  SelectedDescriptor := AskQuestionWithLookup('Descriptor Code','Please select the
Descriptor Code',1010,);
  TheLastCostDate := GetDateFromTable('DescriptorMaster','Lastcostdate',
'DescriptorCode',SelectedDescriptor);
  Showmessage('The Last Cost Date is ' + datetostr(TheLastCostDate));
End.

```

Note how we converted TheLastCostDate to a string value. This is because a Showmessage will only accept String Values

You could try and create your own to (say) get the Order Date from a selected Sales order

16.4.2.5 GetDoubleFromTable

This will access the Ostendo Database and get the numeric value including decimals of a field in a nominated record. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
TableName: The name of the Ostendo Table
FieldName: The Field Name within the Table containing the number
KeyField: The Field against which you are identifying the specific record

KeyValue: The Specific record identity within the Key Field

In this exercise we will access the Ostendo Database and get the '**Stock on Hand**' quantity from a selected Item record

```
// Define the Variable
Var
  SelectedItem: String;
  StockOnHand: Double;
// You can then get the quantity using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  StockOnHand := GetDoubleFromTable('ItemMaster','OnHandQty','Itemcode'
,SelectedItem);
  Showmessage('Stock on hand for Item ' + SelectedItem + ' is ' +
floattostr(StockOnHand));
End.
```

16.4.2.6 GetCost

This will access the Ostendo Database and get the Cost of the selected Item, Descriptor, or Labour Code. The Cost for Items and Descriptors is related to the Cost Method defined in System Settings.

The elements that make up this function are:

Variable: The defined variable against which the result will be held.
Code Type: Descriptor, Item or Labour
Code: The actual Code of the Descriptor, Item or Labour

The following example will get the cost of a selected Descriptor and will display it for your information

```
// Define the Variable
Var
  SelectedDescriptor: String;
  DescriptorCost1: Double;
// You can then get the cost using
Begin
  SelectedDescriptor := AskQuestionWithLookup('Descriptor Code','Please select the
Descriptor Code',1010,);
  DescriptorCost1 := GetCost('Descriptor',SelectedDescriptor);
  Showmessage('Descriptor ' + SelectedDescriptor + ' has a Unit Cost of $' +
floattostr(DescriptorCost1));
End.
```

16.4.2.7 GetStdBuyPrice

This will access the Ostendo Database and get the Standard Buy Price of the selected Item, Descriptor, or Labour Code. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
Code Type: Descriptor, Item or Labour
Code: The actual Code of the Descriptor, Item or Labour

In this exercise we will access the Item Master and extract the Standard Buy Price against an Item

```

// Define the Variable
Var
  SelectedItem: String;
  PartBuy1: Double;
// You can then get the Buy Price using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  PartBuy1 := GetStdBuyPrice('Item',SelectedItem);
  Showmessage('The Item Buy Price is $' + floattostr(PartBuy1));
End.

```

16.4.2.8 GetStdSellPrice

This will access the Ostendo Database and get the Standard Sell Price of the selected Item, Descriptor, or Labour Code. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
Code Type: Descriptor, Item or Labour
Code: The actual Code of the Descriptor, Item or Labour

In this exercise we will access the Item Master and extract the Standard Sell Price against an Item

```

// Define the Variable
Var
  SelectedItem: String;
  PartSell: Double;
// You can then get the Sell Price using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  PartSell := GetStdSellPrice('Item',SelectedItem);
  Showmessage('The Item Sell Price is $' + floattostr(PartSell));
End.

```

16.4.2.9 GetSQLResult

This will allow you to run a Query and then populate the result in a Variable. The elements that make up this function are:

Variable: The defined variable against which the result will be held.
SQL Statement: The Query

In this exercise we will access the Item Master table and carry out a count of the number of records

```

// Define the Variable to be populated by the selected Query result
Var
  SelectedItem: String;
  QueryResult: String;
// You then specify the Query using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  QueryResult := GetSQLResult('Select OnHandQty from ItemMaster where ItemCode = '
+ SelectedItem + '');
  Showmessage('Item ' + SelectedItem + ' has ' + QueryResult + ' On Hand');
End.

```

For more information on SQL syntax see [ExecuteSQL](#) later in this series of exercises

16.4.3 Data from a Spreadsheet

In this section we will:

- Import data from a single Cell and place its content into an existing Ostendo record.
- Import multiple records each containing multiple fields and add these to an Ostendo Table

Before we begin we should look at the following pre-defined Functions

```
LoadSpreadsheet
SSGetCellText
SSGetColumnCount
SSGetRowCount
InsertRecord
```

16.4.3.1 LoadSpreadsheet

This will load an Excel file into memory. The data can then be used by the remaining functions in this section. The element that makes up this function is

FileName: The full path of the document. Note: This must be an xls type of spreadsheet.

To see this in action you should first create a spreadsheet with (say) the following data. Save it as [ScrapCodes.xls](#). We will use this data throughout this short exercise

ScrapCode1	The First Scrap Code
ScrapCode2	The SecondScrap Code

We will point to this spreadsheet in the following exercises

16.4.3.2 SSGetCellText

This function allows you to get information from a selected Cell in a Spreadsheet. When using this function you should first place the Spreadsheet into memory using the LoadSpreadsheet function described above. The elements that make up this function are:

Col: The specific Column in the Spreadsheet
Row: The specific Row in the Spreadsheet
SheetIndex: The specific Sheet Number in the Spreadsheet

In this exercise we will select the above Spreadsheet and then select data from Cell located at Column 1 Row 1. You should note that Column 1 - Row 1 in the spreadsheet is defined as 0,0 in the function

```
Var
CellData : String
Begin
LoadSpreadSheet('c:\ScrapCodes.xls');
CellData := SSGetCellText(0,1);
Showmessage(CellData);
End.
```

Note: Column and/or Row 0 in the script refers to the first Column or Row in the spreadsheet. In the above example therefore it will extract data from the First Column and the Second Row in the spreadsheet

16.4.3.3 SSGetColumnCount

This function allows you to get information about how many columns are being used in a defined Spreadsheet. You need to know this if you are importing multiple records. When using this function you should first place the Spreadsheet into memory using the LoadSpreadsheet function. A single element is used in this function that defines the Sheet Number within the Spreadsheet that is being interrogated:

SheetIndex: The specific Sheet Number in the selected Spreadsheet

Using the above spreadsheet we will count the number of columns being used

```
begin
  LoadSpreadSheet('c:\ScrapCodes.xls');
  showmessage('Number of Columns is ' + intToStr(SSGetColumnCount(0)));
end.
```

16.4.3.4 SSGetRowCount

This function allows you to get information about how many rows are being used in a defined Spreadsheet. When using this function you should first place the Spreadsheet into memory using the LoadSpreadsheet function. A single element is used in this function that defines the Sheet Number within the Spreadsheet that is being interrogated:

SheetIndex: The specific Sheet Number in the selected Spreadsheet

Using the above spreadsheet we will count the number of rows being used

```
begin
  LoadSpreadSheet('c:\ScrapCodes.xls');
  showmessage('Number of rows is ' + intToStr(SSGetRowCount(0)));
end.
```

16.4.3.5 Import a single Cell and Update an Ostendo record

In this exercise we will get the content of a single Cell and update (say) the AdditionalField_1 field of an Item with that content. This will use another function (**ExecuteSQL** - described later) that will update the selected record

```
Var
  CellData : String;
  SelectedItem : String;
Begin
  LoadSpreadSheet('c:\ScrapCodes.xls');
  CellData := SSGetCellText(0,1);
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code'
,1004,);
  ExecuteSQL('Update ItemMaster set AdditionalField_1 = "' + CellData + '" where
ItemCode = "' + SelectedItem + '"');
  Showmessage('Item Number ' + SelectedItem + ' updated');
End.
```

Run the Script and you will see that the content of the Cell will update **AdditionalField_1** of the

selected record

16.4.3.6 Import and create multiple Ostendo records

In this exercise we will use the [ScrapCodes](#) Spreadsheet, plus the [SSGetContentRowCount](#), [SSGetCellText](#) and [InsertRecord](#) (described later) functions to create Ostendo records.

```

Var
  RowCount : Integer;
  x : Integer;
begin
  LoadSpreadSheet('c:\ScrapCodes.xls');
  RowCount := SSGetContentRowCount;
  for x := 0 to RowCount -1 do
  Begin
    InsertRecord('ScrapCodes',
      'ScrapCode=' + SSGetCellText(0,x,0) + #13 +
      'ScrapDescription=' + SSGetCellText(1,x,0) + #13 +
      'CostCentre=');
  End;
  Showmessage('Scrap Codes Added');
end.

```

If you run this script then the two records in the spreadsheet will now appear as Scrap Codes.

16.4.4 Processing Functions

Up to now you have simply enquired on the database and returned a value. In this section we will use many of the functions you've already used to get data and do something with that data

16.4.4.1 Execute SQL

The ExecuteSQL function allows for a wide-range of activities ranging from inquiring on the database to adding, deleting, or updating records. In this exercise we will extend the [AskQuestionWithLookup](#) function and copy the Item Code from the selected record to [AdditionalField_1](#) of the same record

```

// Define the Variable
Var
  SelectedItem: String;
// You can then ask the question to answer SelectedItem using
Begin
  SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code',
  1004,);
  ExecuteSQL('Update ItemMaster set AdditionalField_1 = "' + SelectedItem + '" where
  ItemCode = "' + SelectedItem + '"');
  Showmessage('Item ' + SelectedItem + ' Updated');
End.

```

In looking at the ExecuteSQL statement you should note that the scripting requires the whole SQL statement to be contained within single Quotes. However, three possible conditions may arise that results in three SQL syntax variations.

1. In the following Query we wish to copy field ITEMCODE to field ADDITIONALFIELD_1 of the same record. As the source of ITEMCODE is within the same record being referenced then no additional single quote marks are required. For example:

```
ExecuteSQL('Update ItemMaster set ADDITIONALFIELD_1 = ITEMCODE');
```

2. You may wish to insert a specific value from within the Query itself. If you had the following

```
ExecuteSQL('Update ItemMaster set ADDITIONALFIELD_1 = Jim Gold');
```

It would be rejected because **Jim Gold** is not a recognised field name in the ItemMaster record. Therefore, you may consider enclosing **Jim Gold** within single quotes to give

```
ExecuteSQL('Update ItemMaster set ADDITIONALFIELD_1 = 'Jim Gold');
```

However, this would also be rejected because the single quote after the = sign tells the Script routine that this is the end of the query. Therefore you need to surround **Jim Gold** with further Quote marks to give

```
ExecuteSQL('Update ItemMaster set ADDITIONALFIELD_1 = "Jim Gold"');
```

Ostendo now recognises the 'double Quote Marks' as starting a 'defined value' field. An additional thing to note is that the Quote Marks are all single Quotes and NOT Speech Marks

3. The third variation is where you are inserting a **Variable** or **Constant** from within the script. We have already used the 'Two Quote Mark' option to cover a defined value. Therefore, in this instance we will use THREE single Quotes. For example

Var

```
SelectedItem: String;
```

Begin

```
SelectedItem := AskQuestionWithLookup('Item Code','Please select the Item Code',
1004,);
```

```
ExecuteSQL('Update ItemMaster set AdditionalField_1 = ""SelectedItem"');
```

Once again there is a problem in that - after the = sign - there are two single Quotes indicating that a 'Defined Value' field will follow.....but it is followed by a single Quote (3 quotes in total) and the final one that signifies the end of the Query. Therefore this type of data also requires a (Space) + (Space) both before and after the Variable and should read as follows:

```
ExecuteSQL('Update ItemMaster set AdditionalField_1 = " + SelectedItem + "');
```

If you run this script it will update ALL items with the selected Item Code as you have not specified the specific record to which this action is to take place. Therefore the Query would read

```
ExecuteSQL('Update ItemMaster set AdditionalField_1 = " + SelectedItem + " where
ItemCode = " + SelectedItem + "');
```

There are many other uses for ExecuteSQL and you will come across a few of them in the following scripts

16.4.4.2 InsertRecord

This function will create one or more lines in any table in Ostendo. The elements that make up this function are:

TableName: The table into which the record is being inserted

Mappings: You should identify the mappings of what is to be inserted. The mappings define the fields in the record being create along with the Value of each field. You should also note that each field entry is separated by **#13**. The script routine recognises this as

the end of this field and applies a 'carriage return' to commence the next field

In this exercise we will add a Scrap Code to the Assembly Scrap Code Table. Go to [General>Reports](#) in Ostendo and select '[Full Listing of Tables](#)'. In the drop-down list 'check' 'ScrapCodes' and print it back to your screen you will see that the ScrapCodes Table has the following fields

ScrapCode - We will enter '**Broken**'
ScrapDescription - We will enter '**Item Broken**'
CostCentre - As this is optional we will leave this blank

The script would therefore look like this:

```
Begin
  InsertRecord('ScrapCodes',
    'ScrapCode=Broken' + #13 +
    'ScrapDescription=Item Broken' + #13 +
    'CostCentre=');
  Showmessage('Scrap Code Added');
End.
```

If you run this script you will see that a new entry will appear in screen [Assembly>Settings>Scrap Codes](#)

16.4.4.3 Progress Bar

If you are importing large files then you may wish to use the Progress Bar which displays the progress of the records being imported. This uses four Functions

ShowProgress - Commences the Progress Bar display
UpdateProgress - This tells the routine at what frequency to refresh the Count Bar
UpdateProgressCaption - The allows you to amend the Caption relative to the status of that progress
EndProgress - This terminates the Progress Bar display

The following example pulls these four functions together and shows you how the Progress Bar works. In this example we have defined a ProgressCount value rather than using the Spreadsheet's RowCount value

```
Const
  ProgressCount = 2000;
Var
  x: Integer;
begin
  ShowProgress('My Progress Bar',ProgressCount);
  For x := 1 to ProgressCount do
  Begin
    if x >= (ProgressCount / 2) then
    Begin
      UpdateProgressCaption('Getting There');
    end;
    UpdateProgress(x);
  end;
  EndProgress;
  Showmessage ('Progress Display Completed');
End.
```

16.4.4.4 Run

This function allows you to run any file from within the script. The element that makes up this function is:

FileName: The full path of the file

Try this simple exercise to run Notepad

```
Begin
  Run('Notepad.exe');
End.
```

16.4.4.5 SetScreenParameter

This function allows you identify specific parameters to be run in conjunction with function RunSystemAction. This function has three variants.

KEYFIELD - The Identity of the Key field in the referenced screen
KEYVALUE - The specific value of the KEYFIELD
TABINDEX - (0 = List, 1 = Detail)

For more details refer to the next function (RunSystemAction)

16.4.4.6 RunSystemAction

This function allows you to run any menu-listed screen in Ostendo. The elements that make up this function are:

Category: The Main Menu Item in Ostendo (Example: Inventory, Purchasing, etc)

SystemAction: The specific screen you wish to run that resides under the above Menu

For example: If you wish to run the Customer screen then the entry will be.

```
Begin
  runsystemaction('Sales','Customers');
End.
```

The above function simply opens the relevant screen. However you may wish it to open on a specific record in the screen and show the 'Details' tab rather than the List tab.

To do this you can use the additional Function **SetScreenParameter**. This has three variants:

KEYFIELD - The Identity of the Key field in the referenced screen
KEYVALUE - The specific value of the KEYFIELD
TABINDEX - (0 = List, 1 = Detail)

The following Exercise shows how this is collated into a complete script

```
Var
  TheCust : String;
begin
  TheCust := AskQuestionWithLookup('Customer','Please select the Customer',1015,);
  RunSystemAction('Sales','Customers');
  SetScreenParameter('KEYFIELD=CUSTOMER');
  SetScreenParameter('KEYVALUE=' + TheCust +);
  SetScreenParameter('TABINDEX=1');
end.
```

If you run this script it will open up the Customer Screen in the **'Detail'** tab for the selected Customer

16.4.5 Run Standard Ostendo routines via Scripting

The following exercises allow you to include standard Ostendo routines from within a Script

16.4.5.1 Create a Job Order

Two functions combine to let you create a Job Order and add Order Lines to it.

- InsertJob
- InsertJobOrderLine

16.4.5.1.1 InsertJob

This will create a Job Header record. As a precursor to entering the parameters you would, of course, ask questions, etc as described earlier in these exercises.

The elements that make up this function are:

OrderNumber: The Job Order Number. Note: If the Job Type's Order Numbering is automatic then this should be left blank.

Order Date: The date the Order was raised. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

RequiredDate: The date the Order is required. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

JobType: The Job Type as defined in Jobs>Settings>Job Types.

Customer: The Order Customer

Description: Description of the Order. Can be blank.

Order Notes: Extended Notes against the Order. Can be blank

Customer PO: The Customer's Purchase Order. This must have an entry if the Customer's Master record is flagged as 'Purchase Order Mandatory'

CustomerAsset: The Identity of the Customer Asset is the Job Style of the Order Type is defined as 'Customer Asset'

UseTemplate: Set to True if you wish to create the Job Order using a Job Template..

Template: If the previous parameter is 'True' then a valid Template should be entered here

EstimatedDuration: The estimated duration that the Job will take. Can be zero

DurationScale: The scale of the duration. The options are 'Minutes' or 'Hours',

ProjectName: If the Job Order is linked to a Project then enter the Project ID here

In this exercise we will use the DEMO database and create a Job Order for Customer **'Jim Gold & Co Ltd'**

Var

TheJobOrderNumber: **String**;

Begin

TheJobOrderNumber := InsertJob(",date, strtodate('01/07/2008'),'Progress','Jim Gold & Co Ltd','Phone Order',' ',',',',False,',0,'Hours');

Showmessage('The Job Order created is ' + TheJobOrderNumber);

End.

16.4.5.1.2 InsertJobOrderLine

This will add Lines to an existing Job Order. You can add these separately to an existing Order by quoting the specific Job Order Number or you can do this after creating the Job Order within the same script and referring to the same Variable.. The elements that make up a Job Order Line are:

OrderNumber: The Job Order Number.that must already exist

CodeType: The type of line being added. The options are Item Code, Descriptor Code, Labour Code, Kitset Code, or Catalogue Code.

LineCode: The identity of the line being added.

LineQty: The Order quantity of the line being added

OverridePrice: Enter 'True' if the next field has an override Price. Enter 'False' if you are using the Base Price

UnitPrice: If the previous field is 'True' then enter the override Price

The following exercise combines the creation of a single Job Order Line immediately after creating the Job Header.

Var

TheJobOrderNumber: **String**;

Begin

```
TheJobOrderNumber := InsertJob(",date, strtodate('01/07/2008'),'Progress','Jim Gold & Co Ltd','Phone Order',",",",False,",0,'Hours');
```

```
InsertJobOrderLine(TheJobOrderNumber,'Item Code','100-2000',3,False,0);
```

```
Showmessage('Job Order ' + TheJobOrderNumber + ' created and Line Added');
```

End.

16.4.5.2 Create a Sales Order

Two functions combine to let you create a Sales Order and add Order Lines to it.

- InsertSalesOrder
- InsertSalesOrderLine

16.4.5.2.1 InsertSalesOrder

This will create a Sales Header record. As a precursor to entering the parameters you would, of course, ask questions, etc as described earlier in these exercises.

The elements that make up this function are:

OrderNumber: The Sales Order Number. Note: If the Sales Type's Order Numbering is automatic then this should be left blank.

OrderType: The Sales Type as defined in Sales>Settings>Sales Types.

Order Date: The date the Order was raised. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

RequiredDate: The date the Order is required. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

Customer: The Order Customer

Description: Description of the Order. Can be blank.

PurchaseOrder: The Customer's Purchase Order. This must have an entry if the Customer's Master record is flagged as 'Purchase Order Mandatory'

OrderNotes: Extended Notes against the Sales Order Header. This can be blank

In this exercise we will use the DEMO database and create a Sales Order for Customer **Jim Gold**

& Co Ltd'**Var**TheSONumber: **String**;**Begin**

```
TheSONumber := InsertSalesOrder(", 'CounterSales', date, strtodate('01/07/2008'), 'Jim
Gold & Co Ltd', 'Fax Order', ", ");
```

```
Showmessage('The Sales Order created is ' + TheSONumber);
```

End.

16.4.5.2.2 InsertSalesOrderLine

This will add Lines to an existing Sales Order. You can add these separately to an existing Order by quoting the specific Sales Order Number or you can do this after creating the Sales Order within the same script and referring to the same Variable.. The elements that make up a Sales Order Line are:

OrderNumber: The Sales Order Number, which must already exist

CodeType: The type of line being added. The options are Item Code, Descriptor Code, Kitset Code, or Catalogue Code.

LineCode: The identity of the line being added.

LineQty: The Order quantity of the line being added

OverridePrice: Enter 'True' if the next field has an override Price. Enter 'False' if you are using the Base Price

UnitPrice: If the previous field is 'True' then enter the override Price

The following exercise combines the creation of a single Sales Order Line immediately after creating the Sales Order Header.

VarTheSONumber: **String**;**Begin**

```
TheSONumber := InsertSalesOrder(", 'CounterSales', date, strtodate('01/07/2008'), 'Jim
Gold & Co Ltd', 'Fax Order', ", ");
```

```
InsertSalesOrderLine(TheSONumber, 'Item Code', '100-2000', 10, False, 0);
```

```
Showmessage('Sales Order ' + TheSONumber + ' and Line Added');
```

End.**16.4.5.3 Create a Purchase Order**

Two functions combine to let you create a Purchase Order and add Order Lines to it.

- InsertPurchaseOrder
- InsertPurchaseOrderLine

16.4.5.3.1 InsertPurchaseOrder

This will create a Purchase Header record. As a precursor to entering the parameters you would, of course, ask questions, etc as described earlier in these exercises.

The elements that make up this function are:

OrderNumber: The Purchase Order Number. Note: If the Purchase Type's Order Numbering is automatic then this should be left blank.

OrderType: The Purchase Order Type as defined in Purchasing>Settings>Purchase Types.

Order Date: The date the Order was raised. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format

dd/mm/yyyy) into a date field

RequiredDate: The date the Order is required. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

Supplier: The Order Supplier

Description: Description of the Order. Can be blank.

Order Notes: Extended Notes against the Order. Can be blank

In this exercise we will use the DEMO database and create a Purchase Order from Supplier '**Bruce Wilson**'.

Var

ThePONumber: String;

Begin

ThePONumber := InsertPurchaseOrder(", 'Standard', date, strtodate('01/07/2008'), 'Bruce Wilson', 'Urgent Order', ");

Showmessage('The Purch Order created is ' + ThePONumber);

End.

16.4.5.3.2 InsertPurchaseOrderLine

This will add Lines to an existing Purchase Order. You can add these separately to an existing Order by quoting the specific Purchase Order Number or you can do this after creating the Purchase Order within the same script and referring to the same Variable.. The elements that make up a Purchase Order Line are:

OrderNumber: The Purchase Order Number which must already exist

CodeType: The type of line being added. The options are Item Code, Descriptor Code, or Catalogue Code.

LineCode: The identity of the line being added.

LineQty: The order quantity of the line being added

OverridePrice: Enter 'True' if the next field has an override Price. Enter 'False' if you are using the Base Price

UnitPrice: If the previous field is 'True' then enter the override Price

The following exercise combines the creation of a single Purchase Order Line immediately after creating the Purchase Order Header.

Var

ThePONumber: String;

Begin

ThePONumber := InsertPurchaseOrder(", 'Standard', date, strtodate('01/07/2008'), 'Bruce Wilson', 'Urgent Order', ");

InsertPurchaseOrderLine(ThePONumber, 'Item Code', '100-2000', 3, False, 0);

Showmessage('Purchase Order ' + ThePONumber + ' and Line Added');

End.

16.4.5.4 Create an Assembly Order

There are four functions relating to creating an Assembly Order. In its basic form you can use InsertAssembly to generate an Order using the parent Item's Bill Of Material. Having created the Order you can then add additional Assembly Lines, Steps and Co/Bi-Products

16.4.5.4.1 InsertAssembly

This will get the Item's Bill Of Material (if one exists) and create the Assembly Order for the entered quantity. As a precursor to entering the parameters you would, of course, ask questions covering the Parent Item Number and Quantity required using parameters as described earlier in these exercises.

The elements that make up this function are:

OrderNumber: The Assembly Order Number. Note: If Order Numbering is automatic (Assembly Rules) then this should be left blank.

Order Date: The date the Order was raised. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

RequiredDate: The date the Order is required. You can enter the word date to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

ItemCode: The parent Item for the Assembly Order. You should note that upon generation of the Assembly Order Ostendo will check to see if a BOM exists for this Item and, if so, add the components and routing from the BOM.

Qty: The quantity being ordered

Source Type: The source of the order (Example: Manual, Sales, Jobs)

Source Reference: The number of the Source Order. This can be blank for Manual Source Type

SourceID: Other source such as Drawing Number. Can be left blank

SourceName: Other source Name. Can be left blank

CreateDefaultStep: Set to True if you wish to add the default Assembly step to the Assembly Order. If set to False then you should also run function InsertAssemblyStep to add the relevant steps

In this exercise we will use the DEMO database and an order for Parent Item **1105-2184**.

Var

TheOrderNumber: **String**;

Begin

TheOrderNumber := InsertAssembly(",date, strtodate('01/07/2008'),'1105-2184',1, 'Manual',",0,'To Stock',False);

Showmessage('The Assembly Order created is ' + TheOrderNumber);

End.

If you run this script you will see that it has also generate Order Lines and Steps from the Parent Item's BOM

16.4.5.4.2 InsertAssemblyStep

This will add an Assembly Step an existing Assembly Order. You can add this after an Order has been generated by quoting the specific Assembly Order Number or you can do this after creating the Assembly Order within the same script and referring to the same Variable. In this instance the Step will be added to any current Steps copied from the BOM. The elements that make up an Assembly Step are:

OrderNumber: The Assembly Order Number.

StepName: The name of the Step. This MUST already exist in

Assembly>Settings>StepNames

StepSequence: Any Integer

Instructions: Extended Instruction on the Step Process

The following exercise adds a Step to an existing Assembly Order.

Var

```
TheAssemblyNumber: String;
TheStepID: String;
```

Begin

```
TheAssemblyNumber := AskMandatoryQuestionWithLookup('Assembly Order','Please
select the Order Number',1082,);
```

```
TheStepID := DisplayData('Select * from StepNames','Step Names','StepName'
,500,1200);
```

```
InsertAssemblyStep(TheAssemblyNumber,TheStepID,20,"");
```

```
Showmessage('Assembly Step ' + TheStepID + ' added to Assembly Order ' +
TheAssemblyNumber +);
```

End.

16.4.5.4.3 InsertAssemblyLine

This will add Lines to an existing Assembly Order. You can add these separately to an existing Order by quoting the specific Assembly Order Number or you can do this after creating the Assembly Order within the same script and referring to the same Variable. In this instance the Line will be added to any current lines copied from the BOM. The elements that make up an Assembly Order Line are:

OrderNumber: The Assembly Order Number which must already exist

Qty: The quantity of the line that is being added

LineNumber: The Assembly Order Line Number to be given to this line

CodeType: The type of line being added. The options are Item Code, Descriptor Code, or Labour Code.

LineCode: The identity of the line being added.

Description: Description of the line being added.

StepName: The Step Name that this line is being added to. The Step must already exist against the Assembly Order.

LineUnit: The Unit of Measure for the Line. This must be a valid Unit of Measure currently held against the LineCode

PositionReference: The position Reference (Example: Location on Drawing that the component appears in this order. Can be left blank.

RunOrSetup: Must have either Run or Setup as an entry

LineNotes: Any required notes. Can be left blank

The following exercise adds a line to an existing Assembly Order.

Var

```
TheAssemblyNumber: String;
```

Begin

```
TheAssemblyNumber := AskMandatoryQuestionWithLookup('Assembly Order','Please
select the Order Number',1082,);
```

```
InsertAssemblyLine(TheAssemblyNumber,3,60,'Item Code','100-2000','Washer-Mild
Steel-8mm','Assembly','Each','Run',);
```

```
Showmessage('Line added to Assembly Order ' + TheAssemblyNumber + );
```

End.

16.4.5.4.4 InsertAssemblyOutput

This function will create additional output records to an existing Assembly Order covering Co-Products and Bi-Products. The elements that make up an Assembly Order Output Line are:

OrderNumber: The Assembly Order Number that must already exist

Output Style: This must be either CoProduct or BiProduct

OutputCode: Must be a valid Item Code

OutputDescription: Description of the Co-Product or Bi-Product.

OutputQty: The quantity of the line in the Assembly Order

OutputUnit: The base unit of the OutputCode

OutputCostPercentage: The percentage of the component costs that this Co-Product or Bi-Product will consume

OutputRecQty: The quantity already received against this line

ScrapQty: The quantity already scrapped against this line

The following exercise adds a Co-Product to an existing Assembly Order.

Var

TheAssemblyNumber: String;

TheCoProduct: String;

Begin

TheAssemblyNumber := AskMandatoryQuestionWithLookup('Assembly Order','Please select the Order Number',1082,);

TheCoProduct := AskMandatoryQuestionWithLookup('Co-Product','Please select the Item',1004,);

InsertAssemblyOutput(TheAssemblyNumber,'CoProduct',TheCoProduct,,"2,'Each',40,0,0);
Showmessage('Co-Product added to Assembly Order ' + TheAssemblyNumber +);

End.

16.4.5.5 Create a Timesheet Batch

Two functions combine to let you create a Timesheet Batch and add Lines to the Timesheet.

- InsertTimesheetHeader
- InsertTimesheetLine

16.4.5.5.1 InsertTimesheetHeader

This will create a Timesheet Batch Header record. As a precursor to entering the parameters you would, of course, ask questions, etc as described earlier in these exercises.

The elements that make up this function are:

Status: The status of the Timesheet. This would normally be 'InProgress' however the status could be 'Updated' if you are adding Timesheet history

Reference: Reference text against the Timesheet. This can be blank

EntryStyle: The entry style which can be one of Employee, Job, Assembly, Non-Charge, or Any

EntrySelection: Mandatory entry that must be related to the Entry Style chosen in the previous field

UseTimeOfDay: If 'True' then any Timesheet Lines will have a start and finish time. If False then the lines will have duration only

TimesheetDate: The date the Timesheet was raised. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

ApprovalStatus: If 'Timesheet Approvals' under Labour Settings is checked then this must have an entry. The entry options are 'Waiting Approval', 'Approved', or 'Approval OnHold'.

ApprovalDate: If the status is 'Approved' then this must contain a date. This can be the word 'date' to denote the system date or you can specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

ApprovalUserName: The Name of a User who has 'Approvals' authority

TimesheetNotes: Additional Notes that can apply to the Timesheet. This can be blank

TimesheetBatchDate: The date of the Timesheet Batch. You can enter the word 'date' to

denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

In this exercise we will create a Timesheet for **'Bob Drum'**

Var

TheBatchNumber: Integer;

Begin

TheBatchNumber := InsertTimesheetHeader('InProgress',", 'Employee','Bob Drum', False,date, 'Approved',date, 'Admin',",date);

Showmessage('Timesheet ' + IntToStr(TheBatchNumber) + ' created');

End.

16.4.5.5.2 InsertTimesheetLine

This will add TimeSheet Lines to an existing Timesheet Batch. You can add these independently to an existing Batch by quoting the specific Batch Number or you can do this after creating the Batch Header within the same script and referring to the same Variable.. The elements that make up a Timesheet Line are:

BatchNo: The Timesheet Batch Number held against the Timesheet Header record.

DateWorked: The date the work was carried out. You can enter the word 'date' to denote the system date or specify strtodate('01/07/2008') to convert an entered date (Format dd/mm/yyyy) into a date field

OrderType: The type order against which the work was carried out. The options are Job, Assembly, or Non-Charge

ReferenceNumber: the Reference number within the Order Type entered in the previous field. This must exist in Ostendo

Employee: The Employee Name

EmployeeRateScale: Mandatory entry that must be related to a Rate Scale maintained under Labour>Settings

TaskOrStepName: A valid Task if booking against a Job or Step Name if booking against an Assembly Order. Not required for Non-Charge

LabourCode: A valid Labour Code is required.

DayStartTime: If the 'UseTimeOfDay' in the Timesheet Batch header is True then this must have an entry to the format HH:MM

DayEndTime: If the 'UseTimeOfDay' in the Timesheet Batch header is True then this must have an entry to the format HH:MM

HoursWorked: If 'Timesheet Approvals' under Labour Settings is not checked then this must have an entry. The entry is in the form of HH.DD

ChargeStyle: Enter the Charge Style. The options are Chargeable, Warranty, Contract, or Non-Charge

Non-Charge Code: If the Charge Style is not Chargeable then this must have an entry that relates to the Style being booked against

RecordedNotes: Additional Notes that can apply to the Timesheet Line. This can be blank

CopyToHistory: Can be True or False as required

CopyToLines: Can be True or False as required

CopyToInvLine: Can be True or False as required

LineStatus: Can be InProgress or Updated

The following exercise combines the creation of a single Timesheet Line record immediately after creating the Timesheet Batch Header. It uses the base DEMO database for the source data.

Var

TheBatchNumber: Integer;

Begin

TheBatchNumber := InsertTimesheetHeader('InProgress',", 'Employee','Bob Drum'


```
,False,date,'Approved',date,'Admin',",date);
InsertTimesheetLine(TheBatchNumber,date,'Job','JOB400002','Bob Drum','STD','
BackupRestore','LAB-ASSEMBLY',0,0,5,'Chargeable',",",False,False,False,'InProgress');
Showmessage('Timesheet ' + IntToStr(TheBatchNumber) + ' created and Line added');
End.
```

16.4.5.6 Inventory Replenishment Run

This function allows you to run the Inventory Replenishment Routine. The elements that make up this function are:

Horizon: The scheduling option relating to the Horizon Days. The options are Leadtime or Fixed

HorizonDays: The number of days relating to the previous option

ItemFrom: Enter the start Item Code for the run criteria. Not required if Multi-Level is False.

ItemTo: Enter the end Item Code for the run criteria. Not required if Multi-Level is False.

CategoryFrom: Enter the start Category Code for the run criteria. Not required if Multi-Level is False.

CategoryTo: Enter the end Category Code for the run criteria. Not required if Multi-Level is False.

ABCFrom: Enter the start ABC Code for the run criteria. Not required if Multi-Level is False.

ABCTo: Enter the end ABC Code for the run criteria. Not required if Multi-Level is False.

ExcludeForecast: This should be either True or False

MultiLevelExplosion: This should be either True or False

An alternative to running this from within a script is to run function RunSystemAction and enter the parameters using the standard Ostendo screen

In this exercise we will run a simple Replenishment

```
Begin
RunInventoryReplenishment('Fixed',7,'100-2000','100-2006',",",",",False,False);
End.
```

Try adding the **AskQuestionWithLookup** function to enter a range of Product Codes

16.4.5.7 Run an Ostendo Report

You can run any Report from within a script by using Function **OstendoReport**. The elements that make up this function are:

Report Name: The Name of the Report as known by Ostendo

Device Index: Specify how the report is to be output. The options are:

- 0 or Blank = Standard output option selection panel
- 1 = Select a Printer for printing the report
- 2 = Immediately print on Default Printer
- 3 = Email Direct
- 4 = Email with Dialog
- 5 = Output to Screen

CondValuesCommaText: If the Report has parameters then this can either be left blank and the Parameter Entry screen will be presented or you can enter the Parameters here to automatically prefill the Parameter Values. When used the following format must be adopted:

```
'Condition1=Value1' + #13 + 'Condition2=Value2'
```

Note: The function recognises #13 as a 'Carriage Return'

EmailAddress: Only required if being sent by Email
CCAddress: Optionally required if being sent by Email
BCCAddress: Optionally required if being sent by Email

In this exercise we will print the '**Location Listing**' Report

Begin

```
OstendoReport('Item Summary Listing',0,
'From Item Category=Fasteners' + #13 +
'To Item Category=Fasteners' + #13 +
'From Item Code=' + #13 +
'To Item Code=' + #13 +
'Exclude Conditions=Yes');
```

End.

Try amending the above where you use the '**AskQuestionWithLookup**' function to select the range of Item Codes

16.4.5.8 Run an Ostendo Analysis View

You can run any Analysis View from within a script by using Function **OstendoAnalysis**. The elements that make up this function are:

Analysis Name: The Name of the Analysis View as known by Ostendo
CondValuesCommaText: If the Analysis View has parameters then this can either be left blank and the Parameter Entry screen will be presented, or you can enter the Parameters as part of the script to automatically prefill the Parameter Values. When used the following format must be adopted:

```
'Condition1=Value1' + #13 + 'Condition2=Value2'
```

Note: The function recognises #13 as a 'Carriage Return'

In this exercise we will run the '**Analysis - Item Listing**' view

Begin

```
OstendoAnalysis('Analysis - Item Listing',
'From Item Category=Electrical' + #13 +
'To Item Category=Electrical' + #13 +
'From Item Code=' + #13 +
'To Item Code=');
```

End.

Try amending the above where you use the '**AskQuestionWithLookup**' function to select the range of Item Categories

16.4.5.9 Run an Ostendo Chart View

You can run any Ostendo Chart View from within a script by using Function **OstendoChart**. The elements that make up this function are:

Chart Name: The Name of the Chart View as known by Ostendo
CondValuesCommaText: If the Chart View has parameters then this can either be left blank and the Parameter Entry screen will be presented, or you can enter the Parameters as part of the script to automatically prefill the Parameter Values. When used the following format must be adopted:

```
'Condition1=Value1' + #13 + 'Condition2=Value2'
```

Note: The function recognises #13 as a 'Carriage Return'

In this example we will run 'Chart - Inventory Values'

Begin

```
OstendoChart('Chart - Inventory Values',
'From Warehouse=Main' + #13 +
'To Warehouse=Secondary' + #13 +
'From Location=' + #13 +
'To Location=' + #13 +
'From Category=' + #13 +
'To Category=');
```

End.

Try amending the above where you use the 'AskQuestionWithLookup' function to select the range of Warehouses

16.4.5.10 Run an Ostendo Pivot View

You can run any Ostendo Pivot View from within a script by using Function **OstendoPivot**. The elements that make up this function are:

Pivot Name: The Name of the Pivot View as known by Ostendo

CondValuesCommaText: If the Pivot View has parameters then this can either be left blank and the Parameter Entry screen will be presented, or you can enter the Parameters as part of the script to automatically prefill the Parameter Values. When used the following format must be adopted:

'Condition1=Value1' + #13 + 'Condition2=Value2'

Note: The function recognises #13 as a 'Carriage Return'

In this exercise we will run 'Pivot - Inventory Transactions'

Begin

```
OstendoPivot('Pivot - Inventory Transactions',
'From Item Category=' + #13 +
'To Item Category=' + #13 +
'From Item Code=AAAAAAA' + #13 +
'To Item Code=WWWWW' + #13 +
'From Date=' + #13 +
'To Date=');
```

End.

Try amending the above where you use the 'AskQuestionWithLookup' function to select the range of Item Categories

16.4.5.11 Run an Ostendo Inquiry

You can run any Inquiry display from within a script by using Function **OstendoInquiry**. The elements that make up this function are:

InquiryName: The Name of the Inquiry as known by Ostendo

CondValuesCommaText: If the Inquiry has parameters then this can either be left blank and the Parameter Entry screen will be presented, or you can enter the Parameters as part of the script to automatically prefill the Parameter Values. When used the following format must be adopted:

'Condition1=Value1' + #13 + 'Condition2=Value2'

Note: The function recognises #13 as a 'Carriage Return'

In this exercise we will run the 'Inquiry - Items' view. This Inquiry does not have any parameters

therefore additional parameter entries are not required

```

Begin
OstendoInquiry('Inquiry - Items');
End.

```

Try amending the above where you use the '**AskQuestionWithLookup**' function to select the range of Items

16.5 Where Scripts are Used

There are many areas and styles where Ostendo Scripts can be run. These are:

- **Main Menu:** Adds the Script Name to the drop-down under 'Custom' on Ostendo's main menu
- **Command Line Script:** This allows you to run a script from the Command Line of your computer. This is very useful if, for example, you wish to run the script as part of your computer's overnight batch process.
- **Desktop Icon Script:** This allows you to run a script from a Desktop Icon without the necessity of starting up Ostendo.
- **Related Menu Script:** Adds the Script Name to the drop-down Menu held against the 'Related' button within specific screens.
- **Order Script:** This allows you to create a Script specific to a preselected Order
- **Screen Data Script:** Related to Master, Order, Receiving, and Invoicing Screens where the action of Adding or Deleting a record or changing any field within the record will automatically run this Script
- **Custom Product Script:** Used in Sales and Job orders to configure a Customer-specific product. The result of the configuration creates an Assembly Order
- **Workflow Script:** These are scripts that can be used to change the appearance and/or Text of a Workflow Object.
- **Report Layout Editor:** Scripts to control many printing options such as printing fields under certain data conditions, etc

16.5.1 Main Menu Script

This adds the Script Name to the drop-down under '**Custom**' on Ostendo's main menu. Selecting this will run the Script. This style of Script can be:

- Available to all
- Restricted to Administrator
- Restricted to specific Users

Examples of the types of script are:

- Updating your existing system with information from Ostendo
- Updating Ostendo with information from your existing system
- Importing Supplier Catalogues
- Send user-defined KPI information via email, or to a Mobile phone

To see how this works you can try the following exercise in which we will get an existing Job that was inadvertently created as an Order and we want to change the status to 'Quote'.

Go into **File>Custom Scripts** and add a new Custom Script called (say) **Job Quote**. 'Check' the 'All Users' radio button to make this script available to all users. In the 'Script' tab enter the following:

```

var
  TheJobNumber: String;
  TheCurrentStatus: String;
  AreYouSure: String;

begin
  TheJobNumber := AskMandatoryQuestionWithLookup('Job Number','Select Job Number'
,1075);
  TheCurrentStatus := GetStringFromTable('JOBHEADER','ORDERSTATUS',
'ORDERNUMBER',TheJobNumber);
  AreYouSure := AskQuestion('Confirm this is the Job','TEXT','Are you sure you wish to
change ' + TheJobNumber + ' to a Quote',[d]Yes,No');
  if AreYouSure = 'Yes' then
    begin
      executeSQL('update JobHeader set OrderStatus = "Quote" where OrderNumber = "" +
TheJobNumber + ""');
      executeSQL('delete from OrderRequirements where DemandOrderType = "Job" and
DemandOrderNumber = "" + TheJobNumber + ""');
    end;
  end.

```

Save and exit the Custom Script screen

You will see a new entry 'Custom' on the top toolbar of Ostendo. If you click on this then your Script 'Job Quote' will be displayed. If you click on this you will be taken through the script after which the Job Status will be updated to 'Quote'

16.5.2 Desktop Icon Script

This allows you to run a script from a Desktop Icon without the necessity of starting up Ostendo. In this exercise we will determine the number of Sales Orders created today.

Go into **Sales>Sales Orders** and create a couple of Orders

Go into **File>Custom Scripts** and add a new Custom Script called (say) **SalesToday**. In the 'Script' tab enter the following:

```

// Define the Variable
Var
  OrdersToday: String;
// You can then get the number of Sales Orders created today
Begin
  OrdersToday := GetSQLResult('Select Count(*) from SalesHeader where OrderDate =
"now");
  Showmessage('There are ' + OrdersToday + ' Orders Today');
End.

```

16.5.3 Command Line Script

This allows you to run a script from the Command Line of your computer and is very useful if, for example, you wish to run the script as part of your computer's overnight batch process.

In this exercise we will use the script created above and run it from the Command Line

Open Notepad on your PC then copy and paste the following

"C:\Program Files\Ostendo\ostendo.exe" STARTUPID:DEMO SCRIPT=SalesToday

Where:

- "C:\Program Files\Ostendo\ostendo.exe" points to the Ostendo executable
- STARTUPID:DEMO defines the Database to look at
- SCRIPT=SalesToday defines the script name within the Database

'Save' this as a .bat file (for example [SalesToday.bat](#)) directly under 'C' drive

To run the .bat file from a Command Line click on 'Start' on your Windows screen and select 'Run'. On the presented panel enter 'CMD' and click the 'OK' Button. Go back to the Root Directory (Hint: Enter `cd ..` to go back one level.). You should end up with `C:\>` Type in the .bat file name (Example: [SalesToday](#)) to give `C:\>SalesToday` and hit the 'Enter' key on your keyboard to run your script

To run the script from a Desktop Icon carry out the following. On your Desktop 'Right Mouse' on your existing Ostendo Icon that points to this 'DEMO' Company database and then copy and paste to create another Desktop Icon. On the copied Icon 'Right Mouse' and select 'Properties'. In the 'Target' field extend the target reference to the following

"C:\Program Files\Ostendo\ostendo.exe" STARTUPID:DEMO SCRIPT=SalesToday

and save the changes. If you now double-click on the Icon it will run the script

16.5.4 Related Menu Script

This adds the Script Name to the drop-down Menu held against the 'Related' button within specific screens. The Related script can optionally take key information from the current screen and show specific - related - information.

In addition to using the Ostendo functions described previously this routine uses three additional functions:

- GetSourceFieldValue
- ScreenDataParameter
- RunSystemAction

RunSystemAction and ScreenDataParameter have already been addressed earlier in these exercises.

GetSourceFieldValue

When used in a Related Menu script Ostendo knows the identity of the Main Screen (i.e. Job Order, Sales Order, Purchase Order, etc) and therefore the available data in the Main Screen. This combines with functions [RunSystemAction](#) and [SetScreenParameter](#) to create all the information necessary for generating a Related Menu Script.

The following exercise will pull all this together to create a Related Menu screen showing details of the Customer linked to a selected Job Order

Create a new Script called (say) 'Job Customer'. In the 'Detail' tab 'check' the 'Add to this Screen' checkbox and select 'Job Orders' from the drop-down list in the adjacent field. In the 'Script' tab add the following script

`begin`

```

RunSystemAction('Sales','Customers');
SetScreenParameter('KEYFIELD=CUSTOMER');
SetScreenParameter('KEYVALUE=' + GetSourceFieldValue('CUSTOMER'));
SetScreenParameter('TABINDEX=1'); // This will open up in the Customer 'Detail' screen
rather than the 'List' screen
end.

```

Note: As this has been added to the **'Job Orders'** screen Ostendo knows that the Source Field Values are from the Job Order. Therefore the KEYVALUE of the Related Customer screen is directly linked to the GetSourceFieldValue obtained from the Job Order screen

If you now go into **Jobs>Job Orders** and click on the **'Related'** button. You will find that the **'Job Customer'** Custom Script appears in the drop-down list. If you select this then the script will bring up the Customer 'Detail' Screen relating to this Job Order

16.5.5 Order Script

Order Script

This allows you to create a Script that is run against an Order in the following areas:

- Assembly
- Jobs
- Sales
- Purchase
- POS

and enables you to add extra specific functionality such as:

- Total Order Value discounting based on Order content
- Freight calculations based on Order content
- Order Authorisation Levels (Example: User Purchasing levels)
- Order Margin Control with User-defined Margin levels
- Order Validation and/or Checks
- Workflow actions (Example: send Email regarding this Order)
- Promotions (Example: 3 for price of 2, etc)

This type of script uses all the Ostendo functions described earlier plus:

- GetSourceFieldValue (Described in the previous Exercise)
- OrderScriptRun.

The **OrderScriptValue** enables you to link any scripting activity specifically to this Order. The use of this function has already been covered in the previous ([Related Menu Script](#)) section

The **OrderScriptRun** function is used to annotate if the Order Script has been run, or not. You have the option to prevent any further activity against the Order until this script has been run.

In the following exercise we will create a Sales Order Script that will show that the Order has been accepted and will also update the Tracking Code for the Order and allow the order to continue.

Go into **File>Sales>Settings>Sales Tracking** and add a new Tracking Code called **'Accepted'**

Go into **File>Custom Scripts** and add a new Custom Script called (say) **OrderCust**. 'Check' the **'This is an Order Script'** checkbox. To define if this script must be acknowledged before the Order can proceed you should also 'check' the **'Mandatory'** checkbox.

In the 'Script' tab enter the following:

```
var
  TheOrderNumber: string;
begin
  TheOrderNumber := GetSourceFieldValue('ORDERNUMBER');
  showmessage('Order Number ' + TheOrderNumber + ' Accepted' );
  ExecuteSQL('update SalesHeader set WorkFlowStatus = "Accepted" where
ORDERNUMBER = "' + TheOrderNumber + '"');
  OrderScriptRun(True);
end.
```

The next step is to tell Ostendo that the script is linked to a Sales Order. To do this go into **File>System Configuration>Order Scripts** and create a new record containing the following

Screen: Select 'Sales Orders' from the drop-down
Script Name: Select OrderCust

Now go into **Sales>Sales Orders** and create a Sales Order then add a line to the Order. If you try and pick a line then you will be presented with an error message stating that OrderCust needs to be run'

You will see a new button (**OrderCust**) on the Batch Entry Bar of the Sales Order Lines screen. If you click on this button then the script will be run. This will return the confirmation that it has been run addition to updating the Tracking Code status.

If you now look at the Sales Header you will see that the status has been updated to '**Accepted**'. (You may have to click off the record and back onto it to refresh the screen)

16.5.6 Screen Data Script

This is related to Master, Order, Receiving, and Invoicing Screens where the action of Adding or Deleting a record or changing any field within the record will automatically run the Script to provide a resultant action. For example

- Zero Price Check on Sales Order Lines
- Update Sell Price based on Last receipt Cost
- In Purchasing check for best price from all Suppliers
- Have a pop-up Sales Message appear
- Specify a minimum order quantity
- Show active Promotion when Sales Line Entered

In this exercise we will create a Custom Script that will block any Price or Cost change to an Order Line if the resultant margin falls below the value defined in the System Settings screen.

Go into **File>System Configuration>System Settings** and amend field 'Min Allowable Margin%' to **50**. Now, go to **File>Custom Scripts** and add a new Custom Script called (say) **Margin Check** and 'check' the '**This is a Screen Data script**' checkbox. Click on the '**Script**' tab and add the following script

```
var
  TheMinMargin,TheUnitPrice,TheUnitCost,TheCalcMargin: double;
  TheIntValueofMargin: integer;
begin
  TheMinMargin := GetDoubleFromTable('SYSTEMMASTER','MINMARGINPERCENT',
```



```

'COMPANYACCOUNTINGID','100');
TheUnitCost := GetCost(queryvalue('CODETYPE'),queryvalue('LINECODE'));
TheUnitPrice := strtfloat(queryvalue('ORDERUNITPRICE'));
if (TheUnitPrice <> 0) then
begin
TheCalcMargin := (((TheUnitPrice - TheUnitCost)/ TheUnitPrice) * 100);
if (TheCalcMargin < TheMinMargin) then
begin
TheIntValueofMargin := int(TheCalcMargin * 100);
TheCalcMargin := (TheIntValueofMargin /100);
messagedlg('A ' + floattostr(TheCalcMargin) + ' % margin is below the Company
Minimum of ' + floattostr(TheMinMargin) + ' %',mtinformation,mbOK,0);
end
end
else
begin
if (TheUnitCost > 0) then
Showmessage('This line has a Zero Price with a Cost');
end;
end.

```

Save and exit the Custom Script screen.

The next step is to tell Ostendo that the script is linked to a Sales Order Line. To do this go into **File>System Configuration>Screen Data Scripts** and create a new record containing the following

Screen: Select **Sales Orders** from the drop-down
Table Name: Select **SALESLINES** from the drop-down list
SQL Type: Select **Insert**
Script Name: Select the above script Name

Save and exit

Now go into **Sales>Sales Orders** and create a Sales Order then add a line to the Order. If you amend the sell price against the Item such that the resultant price falls below the above Margin then you will get a message returned and you will be prevented from saving the line.

You should note that the above condition in **File>System Configuration>Screen Data Scripts** only covers when you Insert a Line. You should also create a record that covers when you amend an existing line. For example:

Screen: Select **Sales Orders** from the drop-down
Table Name: Select **SALESLINES** from the drop-down list
SQL Type: Select **Update**
Script Name: Select the above script Name

16.5.7 Custom Product Scripts

These are used in Sales and Job orders to configure a Customer-specific product. This generates a Bill of Material for the specific configuration which can subsequently be converted into an Assembly Order.

A Custom Product Script uses the Functions previously described plus the following functions specifically designed for generating the Custom Product BOM:

- InsertBOMHeader
- InsertBOMStep
- InsertBOMLine
- InsertBOMProperty
- InsertBOMResource
- SetBOMInstructions
- SetBOMLeadTime
- SetBOMRunDuration
- SetBOMSetupDuration
- GetCustomerSellPrice

These are described individually and all pulled together to form a single Custom Product

16.5.7.1 InsertBOMHeader

This function creates a BOM Header for the specific configuration. It is required to define that a BOM is being created. A single description is required as follows:

```
InsertBOMHeader('The BOM Header for this configuration');
```

16.5.7.2 InsertBOMStep

This function will create a Bill of Material Step. The elements that make up this function are:

- StepName:** The name that you are applying to the Step (max 20 chars)
- StepSequence:** A number defining the sequence in which the Step will be carried out
- StepDescription:** A brief description of the Step (max 50 chars)
- StepInstructions:** Unlimited Text defining what happens in this Step

Example:

```
InsertBOMStep('Cut',10,'Cut and Trim all Timber','Check the measurements');
```

16.5.7.3 InsertBOMLine

This function adds a BOM Line to the configuration. It will add a Line to a step in the Bill of Material. The elements that make up this function are:

- StepName:** The Step Name - The step should be created as defined above
- CodeType:** Must be Descriptor, Item or Labour
- LineCode:** The name of the Descriptor, Item or Labour (max 50 chars)
- LineQty:** The quantity required against this line
- LineNumber:** Any Integer to define a Line number
- ScrapPercent:** Any Number (incl decimals). Zero if it does not apply
- RunOrSetup:** Enter either Run or Setup
- PosReference:** Enter a position reference if applicable
- LineInstructions:** Unlimited amount of text to add Instructions

For example

```
InsertBOMLine('Cut','Item','RIMUPANEL1600X600',1,10,0,'Run',"");
```

16.5.7.4 InsertBOMProperty

This function adds Properties to the configured BOM. The elements that make up this function are:

- PropertyName:** The name of the Property (max 20 chars)
 - PropertyValue:** The value being applied to the Property Name (max 50 chars)
-

For example

```
InsertBOMProperty('Length',inttostr(DeskLength));
```

16.5.7.5 InsertBOMResource

This function adds a Resource to a process Step. The elements that make up this function are:

StepName: The Step Name - The step should be created as defined above

ResourceType: Must be either 'ASSET' or 'EMPLOYEE'

ResourceName: The name of the Asset or Employee (max 30 chars)

Example:

```
InsertBOMResource('Cut','Employee','John');
```

16.5.7.6 SetBOMInstructions

This function allows you to define Instructions against the configured BOM Header. The elements that make up this function are:

BOMInstructions: Unlimited amount of text to describe the actions required to make the product

Example:

```
SetBOMInstructions('Cut the 2000mm Rimu Panel down to ' +  
inttostr(DeskLength) + 'mm ');
```

16.5.7.7 SetBOMLeadTime

This function lets you add a Lead Time to the configured BOM Header. The elements that make up this function are:

LeadTime: The number of Days (integer only) required to produce the product

Example:

```
SetBOMLeadTime(2);
```

16.5.7.8 SetBOMRunDuration

This function allows you to add a Run Duration against the configured BOM Header. The elements that make up this function are:

RunDuration: Any Number including decimals

RunDurationScale: Either 'Hours' or 'Minutes'. If nothing entered then Minutes is assumed

Example:

```
SetBOMRunDuration(45,'Minutes');
```

16.5.7.9 SetBOMSetupDuration

This function allows you to define the Setup Duration against the configured BOM Header. The elements that make up this function are:

SetupDuration: Any Number including decimals

SetupDurationScale: Either 'Hours' or 'Minutes'. If nothing entered then Minutes is

assumed

Example:

```
SetBOMSetupDuration(15,'Minutes');
```

16.5.7.10 GetCustomerSellPrice

This is used to derive a Sell Price taking into account the Price Level and Quantity Breaks offered the Customer during Custom creation in Sales or Job Orders. It will access the Ostendo Database and get the Sell Price of the selected Item, Descriptor, or Labour linked to the Customer defined in the Order. The elements that make up this function are:

Variable: The defined variable against which the result will be held.

Code Type: Descriptor, Item or Labour

Code: The actual Code of the Descriptor, Item or Labour

The following example will get the price of the Item and display it for your information. You should first create the Custom Product Item (Example: DOOR') and then add the Item to a Sales Order. The script will then be run within the Order

```
// Define the Variable
Var
SellPrice1: Double;
// You can then get the Price using
Begin
    SellPrice1 := GetCustomerSellPrice('Item','DOOR');
    Showmessage('Item Sell Price is $' + floattostr(SellPrice1));
End.
```

16.5.7.11 A Custom Product Script Example

Let's put the above Custom Product functions together to create a Custom Product Script. In this exercise we will demonstrate how scripting works when configuring Custom products. It is assumed that you are currently in the **DEMO** database as we will be using data currently in that database.

Go into **Inventory>Items** and **'Add'** and Item (say) **'Custom Cut'**. In the **'Detail'** tab set up the following to identify it as a Custom Product:

Default Supply Method: Source on Demand
Sourced By: Custom
Configured By: Rules
'Check' the **'Batch No'** checkbox

Now go into **Assembly>Custom Products** and identify the above **'Custom Cut'** Item as a Custom Product. Click on the **'Script'** tab and add the following

```
Var
WoodType: String;
WoodLength: Integer;
IsEverythingCorrect: String;
TheStandardMarkup: Double;
TheCuttingTime: Double;
ThePanellItem: String;
TheAssemblyTime: Double;
```

```

procedure AskQuestions;
begin
  {Ask all the Questions}
  WoodType := AskQuestion('Please select the type of Wood ','TEXT','Rimu provides a
  better finish and is longer lasting','Pine,[d]Rimu',WoodType);
  WoodLength := AskQuestionNumericRange('Please enter the Length in mm of your ' +
  WoodType + ' Wood ','INTEGER','We can only cut between 30mm - 3000mm in length'
  ,30,3000,1,inttostr(WoodLength));
end;

begin
  AskQuestions;
  {Add the BOM Header}
  InsertBOMHeader;
  {Add a single Step}
  InsertBOMStep('Cutting',10,'Cut the piece','Check the measurements');
  {Standard Labour to Cut the Panel}
  InsertBOMLine('Cutting','Labour','LAB-ASSEMBLY',0.45,10,0,'Run',",");
  {Get the Cost and Add a Markup %}
  LineUnitPrice := LineUnitPrice + (GetCost('Labour','LAB-ASSEMBLY') * 0.45 *
  TheStandardMarkup);
  TheAssemblyTime := 0.45;
  ThePanelItem := 'RP-1600500';
  SetBOMRunDuration(TheAssemblyTime,'Hours');
  SetBOMLeadTime(2);
end.

```

If you now go into **Sales>Sales Orders** and select an open order (or create a new one). Click on the 'Lines' tab in the order and then click the 'Add' button. The cursor will be positioned in the 'Code' field. Click on the 'Spyglass' icon to the right of that field and select 'Custom Cut'. A panel will be presented into which you answer the Custom Script questions. Having answered the questions Ostendo will calculate the cost and arrive at a Sell Price in the Sales order Line.

To generate the Assembly Order click on the 'Related' button to the right of the Dales Order screen and select 'Create Required Assembly Orders'. On the presented screen 'check' the 'Custom Cut' and then click on the 'Generate Orders for Selected requirements' button situated at the bottom-right of the screen. If you go to **Assembly>Assembly Orders** you will see the generated Assembly Order for the Custom Cut.

16.5.8 Workflow Script

These are scripts that can be used to change the appearance and/or Text of a Workflow Object. These include amending the Text, Colour, Hint, Visible/Hide, etc.

The available Functions are:

- SetWorkflowObjectColour
- SetWorkflowObjectGradientColour
- SetWorkflowObjectHint
- SetWorkflowObjectText
- SetWorkflowObjectTransparency
- SetWorkflowObjectVisible
- WorkflowObjectLoadPicture

16.5.8.1 SetWorkflowObjectColour

This function allows you to amend the colour of an Object within a Workflow. The Object must not be currently set to 'Gradient Fill'. This is useful if you wish to have a visual presentation of the status of an object, etc. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID
Colour: See the defined colours in the Workflow Editor

Having created an object in a workflow find out the Object ID by 'right-mousing' on the Object.

The Colour selection is the standard colour preceded with the letters cl. Alternatively you can create your own colour against (say) a Tracking Code and then see - in the database - what the number code is that defines that unique colour. You can then insert the number in place of the colour. For example you can enter either **clAqua** or **13959039**

Begin

```
SetWorkflowObjectColour(4,claqua);  
Showmessage('Colour Updated');
```

End.

16.5.8.2 SetWorkflowObjectGradientColour

This function allows you to amend the Gradient colour of an Object within a Workflow. The Object must be currently set to 'Gradient Fill'. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID
BeginColour: Set the Begin Colour of the Gradient Fill
EndColour: Set the End Colour of the Gradient Fill

Having created the object in a workflow find out the Object ID by 'right-mousing' on the Object.

The Colour selection is the standard colour preceded with the letters cl. Alternatively you can create your own colour against (say) a Tracking Code and then see - in the database - what the number code is that defines that unique colour. You can then insert the number in place of the colour. For example you can enter either **clAqua** or **13959039**

Begin

```
SetWorkflowObjectGradientColour(4,13959039,clnavy);  
Showmessage('Colour Updated');
```

End.

16.5.8.3 SetWorkflowObjectHint

This function allows you to add or amend the Hint held against an Object within a Workflow. Note: A Hint is the line of text that appears when you move the cursor over the Object. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID
Hint: The Text shown when the cursor moves over the Object

Having created the object in a workflow find out the Object ID by 'right-mousing' on the Object.

Begin

```
SetWorkflowObjectHint(4,'This is a Hint');  
Showmessage('Hint Updated');
```

End.

16.5.8.4 SetWorkflowObjectText

This function allows you to add or amend Text that appears within an Object. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID
ObjectText: The Text shown within the Object

Having created the object in a workflow find out the Object ID by 'right-mousing' on the Object.

Begin

```
SetWorkflowObjectText(4,'Workflow');  
Showmessage('Text Updated');
```

End.

16.5.8.5 SetWorkflowObjectTransparency

This function allows you to add or amend the Object so that its transparency can be adjusted if you wish to view other Objects lying behind this one. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID
Transparency: The Transparency value expressed as a percentage

Having created the object in a workflow find out the Object ID by 'right-mousing' on the Object.

Begin

```
SetWorkflowObjectTransparency(4,50);  
Showmessage('Transparency Updated');
```

End.

16.5.8.6 SetWorkflowObjectVisible

This function allows you to make an Object Visible or Invisible. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID
Visible: This can be True or False

Having created the object in a workflow find out the Object ID by 'right-mousing' on the Object.

Begin

```
SetWorkflowObjectVisible(4,False);  
Showmessage('Object now Invisible');
```

End.

16.5.8.7 WorkflowObjectLoadPicture

This function allows you to add or amend the current picture that exists in a Picture Object. The elements that make up this function are:

Object ID: Right-Mouse on the Object in the Workflow to get the ID. Note: The Object must be a Picture Object
Filename: The full path pointing to the Picture

Having created an object in a workflow you need to find out the Object ID by 'right-mousing' on the Object where the ID is displayed

This example will overwrite the current Picture Object. You should note that the Object itself must be a 'Picture' object

Begin

```
WorkflowObjectLoadPicture(6,'c:\House.jpg');  
Showmessage('Picture Updated');
```

End.

16.5.9 Report Layout Editor

Scripting can be used in the Report Layout Editor to control many printing options such as printing fields under certain data conditions, etc. There are two main elements in the Report that controls the script.

- When is the action to take place
- What action is to take place

It is not intended here to give a detailed description of Report Coding but merely to show you that the Code Option exists.

Go into **File>Reporting Configuration>Report and View Developer** and click the 'Add' button. Copy the 'Item Detail Sheet' to your Company Reports folder. Select the copied report and click on the 'Master Settings' tab then on the 'Edit' button. On the report layout scroll down to the 'Child4' Band. You will see a small red triangle (?) in the band. This denotes that it has some code linked to it. If you click on the Band then we will address the linked code as follows

When is the action to take place: If you click on the 'Events' tab to the left of the screen you will see that the 'when' is 'OnBeforePrint' and it relates to a script procedure in the adjacent field. (In this instance **Child4OnBeforePrint**.)

What action is to take place: If you double-click on the **Child4OnBeforePrint**.it will take you to the 'Code' tab and position you at the **Child4OnBeforePrint** procedure. You will see that script provides a visible = True or False depending upon the content of the Notes field.

17 16. Custom Screens

There are two types of Custom Screens that you can create

Custom Data Screens which are designed for fast data entry typically used in POS and SFDC, but could also be used in Stocktakes, Purchase receiving, etc. They comprise of User-Defined entry and display fields whose data source can be Computer Keyboard, Graphical Keyboard, Touch Screen, or Barcode Scanner. All four can be utilised in the same Custom Data Screen.

Data Entry Screens where you can define what data is to be collected and create a multi-line Data entry screen into which the data is entered. The resultant records can be actioned through Ostendo's scripting process

17.1 Custom Data Screens

A Custom Data Screen enables the User to create Custom Screens that provide the following options

Display Types: The Screen itself can be Inquiry only, Data Entry only or a combined Inquiry/Data Entry

Data Entry Styles: You have the option to use a Computer Keyboard, Graphical Keyboard, Touch Screen, or Barcode Scanner. All four can be utilised in the same Custom Data Screen.

Data Options: This can include Open format entry, validated against data within the Custom Data Script, or validated against Ostendo data.

Data Storage Options. The data being entered can be held in temporary storage for 'Batch Posting' into Ostendo or posted immediately - a single record at a time. If the data is held in temporary storage it can be held in non-Ostendo files and subsequently recalled into the data screen.

The screen itself comprises of two main functional areas

Data Entry area

The Data Entry area allows you enter data that is stored in a temporary table for subsequent action within the Custom Data Screen process. This area is primarily designed for Barcode Scanning but can also be used in conjunction with Computer Keyboard input or via a Graphical keyboard displayed in the Graphics Area.

A single Data Entry field is displayed along with an explanation of what is required to be entered. Upon entry and acceptance of that data the next field (and explanation) is displayed....and so on. Upon completion of each field, or optionally the complete record, the content of that record is displayed in the Graphics area in addition to being stored in a temporary table.

Display and Graphics area

The Display and Graphics area uses Ostendo's Graphical developer to create buttons, data display area, graphical keyboard, etc. The objects in this area can be changed by the script to:

- Display Data from the Temporary Table
 - Change the Colour of Objects
 - Change the wording in an object
 - Make the Object invisible/visible
 - Go to another Scheme for entry of related data
-

- Enable selected objects to be touch screen compatible

17.1.1 Creating your first Data Screen Script

In this exercise we will create a simple Data Entry screen that allows us to enter records directly into Ostendo's Standard Units table. We will take a step-by-step approach and describe what is happening at each step. These steps will include:

- Create the Graphical Panel with one data field
- Create the Data Entry routine and see it populate the above field
- Add further fields to the Graphical Panel
- Extend the Data Entry routine to progress through the fields
- Closing the Screen

17.1.1.1 The Data Entry Screen - Step 1

The Graphical Display area provides visibility of what has been entered prior to updating Ostendo. In this first step we will create a single field and demonstrate the data entry routine to populate this single field.

1. Create the Display Graphic area with one data field

Go into your Ostendo graphical Developer (OstDesigner.exe under your Ostendo folder) and create a screen as follows:

Click File>New to create a new screen.

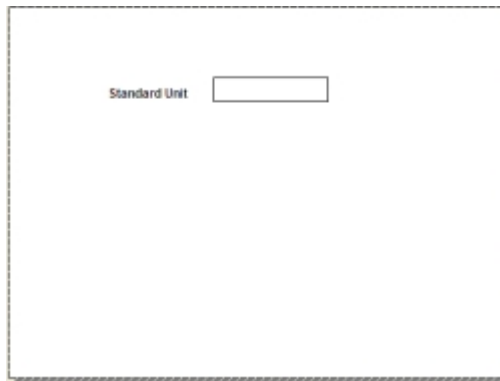
Click the '**Text Tool**' and add a **Text Object** to the screen. In the Inspector panel set the object's characteristics as follows:

Alignment:	Right
Font:	Set Font Size as required
Layout:	Centre
Text:	Make this ' Standard Unit: '

Add another **Text Object** and, in the Inspector panel set the object's characteristics as follows:

Alignment:	Left
Font:	Set Font Size as required
ID:	Note down the Object's ID
Layout:	Centre
Pen:	Pen Style Solid – Black
Text:	Remove any Text

The finished result should look something like this



2. Create the associated Script

In Ostendo go into **File>Custom Scripts** and create a Script called (say) '**FirstDataScreen**'. In the '**Detail**' tab select Style '**Custom Data Screen**'. In the '**Data Screen File**' field point it to the Data Screen you created above.

If you now click on the '**Script**' tab you will find that Ostendo prefills this with three basic procedures used in Data Screen operations. These are:

- DataScreenOnCtrlKey**: which controls actions when nominated keys are pressed along with a Ctrl key
- DataScreenOnValueEntered**: which controls the data entry fields in the screen
- DataScreenObjectClick**: which controls the actions in response to the data entry

We will introduce these procedures in the course of these exercises.

In this first exercise we will create a script that will allow us to enter data and display that data in the object you created above.

To create the data entry field you need to use two functions:

- DataScreenQuestion**: to define the question
- DataScreenShow**; to activate the display of the Question

Therefore, at the end of the basic script add the following

```
begin
  DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT',",",",");
  DataScreenShow;
end.
```

The script should now look like this:

```
procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  {This procedure is fired when the control (Ctrl) key is pressed together}
  {with an alpha key, use it for your own keyboard shortcuts within the script}
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  {This procedure is fired when the enter key is pressed after entering or scanning a value.}
```

```
{after calling DataScreenQuestion you can process the answer or response here}
end;
```

```
procedure DataScreenObjectClick(ObjectID: Integer);
begin
  {This procedure is fired when an object on the top graphical interface
  is clicked, use the ObjectID to identify the object that was clicked}
end;

begin
  DataScreenQuestion(12,'Please Enter a Standard Unit','TEXT',"");
  DataScreenShow;
end.
```

If you run the script then you will see that data entry field, along with the 'Please Enter a Standard Unit' at the bottom of the screen. You can enter data into this field but nothing happens because we have not told the routine what to do with the data. This is where the procedure DataScreenOnValueEntered comes in.

Amend the DataScreenOnValueEntered Procedure to read:

```
procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  {This procedure is fired when the enter key is pressed after entering or scanning a value,}
  {after calling DataScreenQuestion you can process the answer or response here}
  Begin
    DataScreenSetObjectText(12,Value);
  end;
end;
```


What we have done is told the routine to update the Text of Text Object #12 with the content of 'Value' and the content if 'Value' is provided by the answer to function DataScreenQuestion. If you now run the script you should find that if you enter a Value and pres the 'Enter' key of the keyboard the Value will appear in the 'Standard Unit' field.

17.1.1.2 The Data Entry Screen - Step 2

In the previous step you created a single data entry field and displayed the entered value in the main panel. What we are going to do now is add more fields against which we will enter data. This brings up the problem of going through each question in turn and populating the correct field with the entered information.

1. Add extra fields to the Display Graphic area

Go back into your Ostendo Graphic. Note: you can click on the 'Edit' button on the 'Detail' screen of your Custom Script to take you there.

Shift/Click on the two objects to embrace both. Now click on the 'Duplicates' Icon . Position the duplicated object below the above two. Repeat this twice then position and add text to the Objects as shown in the screen below.

Take note of the ID's of the four rectangular Text objects the save the Graphic

2. Create the associated Script

This process involves:

- QuestionIndex: A sequential number, commencing at zero, in which the questions are presented
- Extending the DataScreenQuestion function.
- Action to be carried out as a result of answering each question.
- Additional Variables to temporarily store the entered data

This would produce the following script

Var

```
TheUnit: String;
TheDescription: String;
IsTimeUnit: String;
```

```
procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
```

```
begin
```

```
  {This procedure is fired when the control (Ctrl) key is pressed together}
```

```
  {with an alpha key, use it for your own keyboard shortcuts within the script}
```

```
end;
```

```
procedure (QuestionIndex: Integer; Value: String);
```

```
begin
```

```
  {This procedure is fired when the enter key is pressed after entering or scanning a value,}
```

```
  {after calling DataScreenQuestion you can process the answer or response here}
```

```
  case QuestionIndex of
```

```
    0:
```

```
      Begin
```

```
        TheUnit := Value;
```

```
        If TheUnit = " then
```

```
          Begin
```

```
            DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','A Standard Unit MUST  
be entered',",");
```

```
          end
```

```
        else
```

```
          begin
```

```
            DataScreenSetObjectText(12,Value);
```

```
            DataScreenQuestion(1,'Please Enter a Description','TEXT',",",");
```

```

    end;
  End;
1:
  Begin
    TheDescription := Value;
    If TheDescription = " then
      Begin
        DataScreenQuestion(1,'Please Enter a Description','TEXT','A Description MUST be
entered','');
      end
    else
      begin
        DataScreenSetObjectText(8,Value);
        DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','','Yes,No,');
      end;
    end;
  end;
2:
  Begin
    IsTimeUnit := Value;
    If IsTimeUnit = " then
      Begin
        DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','You must select Yes or
No','Yes,No,');
      end
    else
      begin
        DataScreenSetObjectText(9,Value);
        DataScreenQuestion(3,'Is this a Time Unit?','COMBOBOX','','Yes,No,');
      end;
    end;
  end;
3:
  Begin
    If TimePerHour = " then
      Begin
        DataScreenQuestion(3,'Enter the Time Units','CALC','Entry must be greater than
zero','');
      end;
    else
      Begin
        DataScreenSetObjectText(10,Value);
        DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','','Yes,No,');
    );
  end;
end;
End;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
  {This procedure is fired when an object on the top graphical interface
is clicked, use the ObjectID to identify the object that was clicked}
end;

begin
  DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','','');
  DataScreenShow;

```

end.

Change the Object ID's against each DataScreenSetObjectText to conform to those ID's you have noted above. Copy and Paste this script to overwrite the existing Custom Script

Let's have a look at this script in more detail.

1. Three Variables have been declared – one for each data entry field
2. The DataScreenOnValueEntered procedure contains a QuestionIndex that is used to segregate the Data entry/Data field relationships and the action to be carried out against these.
3. The process commences with the DataScreenQuestion found at the end of the script. This starts at Index 0.
4. Having entered the value into the screen and hitting the enter key it the script progresses to DataScreenOnValueEntered using QuestionIndex 0. If the 'Enter' key is pressed without anything being entered then the same question is asked along with an extended explanation. If an entry is made then the Object in the Graphics screen is populated with the entered value and DataScreenQuestion #1 is asked.
5. This will continue until all questions have been answered

17.1.1.3 Closing the Screen

Up to now you had to click on the 'Close' icon to exit the Data Screen. Let us use (say) the character 'Q' to close the screen. There are two ways in which you can close a screen:

Option 1. As part of the QuestionIndex function you can enquire if the letter 'Q' has been entered and, if so, close the screen. Therefore amend the above Case sections to include the extra line shown below

```

case QuestionIndex of
0:
  Begin
    If uppercase(value) ='Q' then DataScreenClose;
    DataScreenSetObjectText(5,Value);
  End;
end;

```

If you rerun the script then all Text entries will populate the Object until it encounter 'Q' in which case it will run function DataScreenClose.

Try this against the first 'case' statement then remove the line after you see that this works.

Option 1. The previous example showed where you can Close the screen from the data entry field. This example allows you to do this without entering a character into any field

The first of the three procedures of the Data Screen Script is called DataScreenOnCtrlKey. This allows you to define combinations of Ctrl with any alpha character and link an activity with that combination. In this example we will Close the screen when Ctrl-Q is pressed.

Replace the first procedure section with the following

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  {This procedure is fired when the control (Ctrl) key is pressed together}

```

```

    {with an alpha key, use it for your own keyboard shortcuts within the script}
case QuestionIndex of
0:
  Begin
  If Keynumber = 17 then DataScreenClose;
  end;
end;
end;

```

You will see that Keynumber 17 represents the 17th letter of the Alphabet ('Q'). When QuestionIndex 0 is active then clicking **Ctrl-Q** will close the Data Screen.

17.1.1.4 Data Validation

Before we can commence updating Ostendo we need to make sure that the data entries conform to the Database field requirements.

1. Standard Unit

The length of the Standard Unit field 15 characters therefore we need to validate that entry does not exceed this value. Amend the QuestionIndex of 0 section to read

```

  Begin
  TheUnit := Value;
  If uppercase(value) ='Q' then DataScreenClose;
  If TheUnit = " then
  Begin
  DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT',"");
  end
  else
  begin
  If Length(TheUnit) > 15 then
  Begin
  Showmessage('Maximum Length of Standard Units is 15 Characters');
  DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','A Standard Unit MUST
be entered',"");
  end
  else
  begin
  DataScreenSetObjectText(12,Value);
  DataScreenQuestion(1,'Please Enter a Description','TEXT',"");
  end;
end;
End;

```

2. Unit Description

As with Standard Unit the length of the Unit Description field 50 characters therefore we need to validate that entry does not exceed this value. Amend the QuestionIndex of 1 section to read

```

  Begin
  TheDescription := Value;
  If TheDescription = " then
  Begin
  DataScreenQuestion(1,'Please Enter a Description','TEXT',"");
  end

```



```

else
  If Length(TheDescription) > 50 then
    Begin
      Showmessage('Maximum Length of Description is 50 Characters');
      DataScreenQuestion(1,'Please Enter a Description','TEXT','A Description MUST be
entered','');
    end
  else
    begin
      DataScreenSetObjectText(8,Value);
      DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','Yes,No,');
    end;
  end;
end;

```

3. Is Time Unit?

In this check the entry into the screen is either Yes or No from the drop-down list. However this does not prevent the User from manually keying in other data. We therefore need to prevent this from within the script. Amend the QuestionIndex of 2 section to read

```

Begin
  IsTimeUnit := Value;
  If IsTimeUnit = " then
    Begin
      DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','Yes,No,');
    end;
  If ((IsTimeUnit <> 'Yes') and (IsTimeUnit <> 'No')) then
    Begin
      Showmessage ('Is-Time-Unit must be Yes or No');
      DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','You must select Yes or
No','Yes,No,');
    end;
  If (IsTimeUnit = 'Yes') then
    Begin
      DataScreenSetObjectText(9,Value);
      DataScreenQuestion(3,'Enter the Time Units','CALC','','');
    End;
  If (IsTimeUnit = 'No') then
    Begin
      TimePerHour := '1';
      DataScreenSetObjectText(9,Value);
      DataScreenSetObjectText(10,TimePerHour);
      DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','Yes,No,');
    End;
  end;
end;

```

4. Time Per Hour

If the answer to the previous question is Yes then a Time per Hour is required that must be greater than zero. This is checked as follows

```

3:
Begin
  If TimePerHour = " then
    Begin
      DataScreenQuestion(3,'Enter the Time Units','CALC','A Time Unit is a required field',
');

```

```

end;
If (IsTimeUnit = 'Yes') then
begin
  DataScreenSetObjectText(10,Value);
  DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','', 'Yes,No',''
);
end;
end;
end;

```

17.1.1.5 Updating Ostendo

You could use a DataScreenQuestion or a Message Dialog to complete the process. This final step will ask the question “Do you want to Update Ostendo?”. If the answer is ‘Yes’ then create the record and reset the screen ready for the next transaction. If the answer is ‘No’ then just reset the screen.

1. Using a DataScreenQuestion

Continuing with the above Question and Answer process we can now ask a further question using the following script:

```

4:
Begin
  UpdateOstendo := Value;
  If UpdateOstendo = " then
  Begin
    DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','You must
select Yes or No','Yes,No','');
  end;
  If ((UpdateOstendo <> 'Yes') and (UpdateOstendo <> 'No')) then
  Begin
    Showmessage ('Update Ostendo must be Yes or No');
    DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','You must
select Yes or No','Yes,No','');
  end;
  If (UpdateOstendo = 'Yes') then
  Begin
    If IsTimeUnit = 'Yes' then TimeUnit := 1;
    If IsTimeUnit = 'No' then TimeUnit := 0;
    ExecuteSQL('Insert into StandardUnits (StandardUnit, UnitDescription, IsTimeUnit,
TimePerHour) values ('' + TheUnit + '' , '' + TheDescription + '' , ' + IntToStr(TimeUnit) + ' ,
+ floatToStr(TimeHour) + ')');
    DataScreenSetObjectText(8,'');
    DataScreenSetObjectText(9,'');
    DataScreenSetObjectText(10,'');
    DataScreenSetObjectText(12,'');
    TheUnit := '';
    TheDescription := '';
    IsTimeUnit := '';
    TheTimePerHour := '';
    UpdateOstendo := '';
    DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','', '');
  End;
  If UpdateOstendo = 'No' then
  Begin
    DataScreenSetObjectText(8,'');
  End;

```

```

DataScreenSetObjectText(9,"");
DataScreenSetObjectText(10,"");
DataScreenSetObjectText(12,"");
TheUnit := "";
TheDescription := "";
IsTimeUnit := "";
TheTimePerHour := "";
UpdateOstendo := "";
DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT',"");
end;
end;
end;
End;

```

Run the script and confirm that you want to update Ostendo. If you now look in **General>Settings>Standard Units** you will see that the record has been created.

Your final script would be

```

Var
TheUnit: String;
TheDescription: String;
IsTimeUnit: String;
TheTimePerHour: String;
UpdateOstendo: String;
TimeUnit: Integer;
TimeHour: Double;

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  {This procedure is fired when the control (Ctrl) key is pressed together}
  {with an alpha key, use it for your own keyboard shortcuts within the script}
  case QuestionIndex of
    0:
      Begin
        If Keynumber = 17 then DataScreenClose;
      end;
    end;
  end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  {This procedure is fired when the enter key is pressed after entering or scanning a value,}
  {after calling DataScreenQuestion you can process the answer or response here}
  case QuestionIndex of
    0:
      Begin
        TheUnit := Value;
        If uppercase(value) = 'Q' then DataScreenClose;
        If TheUnit = " then
          Begin
            DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','A Standard Unit MUST
be entered',"");
          end
        else
          begin

```

```
    If Length(TheUnit) > 15 then
        Begin
            Showmessage('Maximum Length of Standard Units is 15 Characters');
            DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT',"","");
        end
    else
        begin
            DataScreenSetObjectText(12,Value);
            DataScreenQuestion(1,'Please Enter a Description','TEXT',"","");
        end;
    end;
end;

1:
Begin
    TheDescription := Value;
    If TheDescription = " then
        Begin
            DataScreenQuestion(1,'Please Enter a Description','TEXT','A Description MUST be
entered',"");
        end
    else
        Begin
            If Length(TheDescription) > 50 then
                Begin
                    Showmessage('Maximum Length of Description is 50 Characters');
                    DataScreenQuestion(1,'Please Enter a Description','TEXT',"","");
                end
            else
                begin
                    DataScreenSetObjectText(8,Value);
                    DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX',"','Yes,No','");
                end;
            end;
        end;
    end;

2:
Begin
    IsTimeUnit := Value;
    If IsTimeUnit = " then
        Begin
            DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX',"','Yes,No','");
        end;
    If ((IsTimeUnit <> 'Yes') and (IsTimeUnit <> 'No')) then
        Begin
            Showmessage ('Is-Time-Unit must be Yes or No');
            DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','You must select Yes or
No','Yes,No','");
        end;
    If (IsTimeUnit = 'Yes') then
        Begin
            DataScreenSetObjectText(9,Value);
            DataScreenQuestion(3,'Enter the Time Units','CALC',"","");
        End;
    If (IsTimeUnit = 'No') then
        Begin
```

```

    TheTimePerHour := '1';
    DataScreenSetObjectText(9,Value);
    DataScreenSetObjectText(10,TheTimePerHour);
    DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','', 'Yes,No,');
  End;
end;

3:
Begin
  If TheTimePerHour = " then
    Begin
      DataScreenQuestion(3,'Enter the Time Units','CALC','', '');
    end;
  If (IsTimeUnit = 'Yes') then
    begin
      DataScreenSetObjectText(10,Value);
      TimeHour := StrToFloat(Value);
      DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','', 'Yes,No,')
    );
  end;
end;

4:
Begin
  UpdateOstendo := Value;
  If UpdateOstendo = " then
    Begin
      DataScreenQuestion(4,'Do you want to Update Ostendo?','COMBOBOX','', 'Yes,No,')
    );
  end;
  If ((UpdateOstendo <> 'Yes') and (UpdateOstendo <> 'No')) then
    Begin
      Showmessage ('Update Ostendo must be Yes or No');
      DataScreenQuestion(4,'Do you want to Update
Ostendo?','COMBOBOX','', 'Yes,No,');
    end;
  If (UpdateOstendo = 'Yes') then
    Begin
      If IsTimeUnit = 'Yes' then TimeUnit := 1;
      If IsTimeUnit = 'No' then TimeUnit := 0;
      ExecuteSQL('Insert into StandardUnits (StandardUnit, UnitDescription, IsTimeUnit,
TimePerHour) values ('' + TheUnit + '' , '' + TheDescription + '' , ' + IntToStr(TimeUnit) + ' , '
+ floatToStr(TimeHour) + ')');
      DataScreenSetObjectText(8,");
      DataScreenSetObjectText(9,");
      DataScreenSetObjectText(10,");
      DataScreenSetObjectText(12,");
      TheUnit := "";
      TheDescription := "";
      IsTimeUnit := "";
      TheTimePerHour := "";
      UpdateOstendo := "";
      DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','', '');
    End;
  If UpdateOstendo = 'No' then
    Begin

```

```

    DataScreenSetObjectText(8,"");
    DataScreenSetObjectText(9,"");
    DataScreenSetObjectText(10,"");
    DataScreenSetObjectText(12,"");
    TheUnit := "";
    TheDescription := "";
    IsTimeUnit := "";
    TheTimePerHour := "";
    UpdateOstendo := "";
    DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT',"","");
end;
end;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
    {This procedure is fired when an object on the top graphical interface
    is clicked, use the ObjectID to identify the object that was clicked}
end;

begin
    DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT',"","");
    DataScreenShow;
end.

```

2. Using a Message Dialog

This method uses a standard scripting function called **messagedlg**. Let's first look at this function and we'll then use it in our script.

In Ostendo create a new Custom Script called (say) **Message Dialog**
Copy and paste the following script

```

Var
    TheResult: Integer;

begin
    TheResult := MessageDlg('Click on the Button',mtInformation,1,0);
    ShowMessage(TheResult);
end.

```

If you run the script you will get the number **6** being returned in the Showmessage.

In the MessageDlg line change the number 1 to 2 and re-run the script and see the effect. You will see that a number **7** is returned. The number that is returned reflects the button that has been selected. I.e.

```

OK      = 1
Cancel  = 2
Yes     = 6
No      = 7

```

Try changing the number to any number up to 15 and see what happens.
Try changing the 'mtinformation' to 'mterror', or 'mtwarning'.

We will utilise the returned messagedlg number to define the action to take.

Amending your script.

At this point in time we are going to make a large change to the script you created above therefore you may wish to create a new script and copy the above to it.

Copy the following script and completely replace Case 2, 3 and 4 (we are no longer using Case 4). You will see that we are generating actions based upon the number 6 being returned from the MessageDlg

```

2:
  Begin
  IsTimeUnit := Value;
  If IsTimeUnit = " then
    Begin
    DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','Yes,No,');
    end;
    If ((IsTimeUnit <> 'Yes') and (IsTimeUnit <> 'No')) then
      Begin
      Showmessage ('Is-Time-Unit must be Yes or No');
      DataScreenQuestion(2,'Is this a Time Unit?','COMBOBOX','You must select Yes or
No','Yes,No,');
      end;
      If (IsTimeUnit = 'Yes') then
        Begin
        DataScreenSetObjectText(9,Value);
        DataScreenQuestion(3,'Enter the Time Units','CALC','','');
        End;
      If (IsTimeUnit = 'No') then
        Begin
        TheTimePerHour := '1';
        DataScreenSetObjectText(9,Value);
        DataScreenSetObjectText(10,TheTimePerHour);
        TheAnswer := messagedlg('Do you wish to update Ostendo',mtinformation,3,0);
        if TheAnswer = 6 then
          begin
            ExecuteSQL('Insert into StandardUnits (StandardUnit, UnitDescription, IsTimeUnit,
TimePerHour) values (' + TheUnit + ', ' + TheDescription + ', ' + IntToStr(TimeUnit) + ', '
+ floatToStr(TimeHour) + ')');
            DataScreenSetObjectText(8,");
            DataScreenSetObjectText(9,");
            DataScreenSetObjectText(10,");
            DataScreenSetObjectText(12,");
            TheUnit := ";
            TheDescription := ";
            IsTimeUnit := ";
            TheTimePerHour := ";
            UpdateOstendo := ";
            DataScreenQuestion(0,'Please Enter a Standard Unit','TEXT','','');
          end
        else
          Begin
            DataScreenSetObjectText(8,");
            DataScreenSetObjectText(9,");
            DataScreenSetObjectText(10,");
            DataScreenSetObjectText(12,");
            TheUnit := ";

```

```

        TheDescription := "";
        IsTimeUnit := "";
        TheTimePerHour := "";
        UpdateOstendo := "";
        DataScreenQuestion(0, 'Please Enter a Standard Unit', 'TEXT', "", "");
    end;
end;
end;

3:
Begin
    If TheTimePerHour = " then
        Begin
            DataScreenQuestion(3, 'Enter the Time Units', 'CALC', "", "");
        end;
    If (IsTimeUnit = 'Yes') then
        begin
            DataScreenSetObjectText(10, Value);
            TimeHour := StrToFloat(Value);
            TheAnswer := messagedlg('Do you wish to update Ostendo', mtinformation, 3, 0);
            if TheAnswer = 6 then
                begin
                    ExecuteSQL('Insert into StandardUnits (StandardUnit, UnitDescription, IsTimeUnit,
TimePerHour) values ('" + TheUnit + "', '" + TheDescription + "', ' + IntToStr(TimeUnit) + ','
+ floatToStr(TimeHour) + ')');
                    DataScreenSetObjectText(8, "");
                    DataScreenSetObjectText(9, "");
                    DataScreenSetObjectText(10, "");
                    DataScreenSetObjectText(12, "");
                    TheUnit := "";
                    TheDescription := "";
                    IsTimeUnit := "";
                    TheTimePerHour := "";
                    UpdateOstendo := "";
                    DataScreenQuestion(0, 'Please Enter a Standard Unit', 'TEXT', "", "");
                end
            else
                Begin
                    DataScreenSetObjectText(8, "");
                    DataScreenSetObjectText(9, "");
                    DataScreenSetObjectText(10, "");
                    DataScreenSetObjectText(12, "");
                    TheUnit := "";
                    TheDescription := "";
                    IsTimeUnit := "";
                    TheTimePerHour := "";
                    UpdateOstendo := "";
                    DataScreenQuestion(0, 'Please Enter a Standard Unit', 'TEXT', "", "");
                end;
            end;
        end;
    end;
end;
end;
end;

```


17.1.1.6 Recap

1. *DataScreenQuestion*

The format of this function is:

Format: `DataScreenQuestion(QuestionIndex, Question, EditorType, FullExplanation, ValueList, DefaultValue, LookupIndex);`

This function is used to prompt for data entry. The result of the data entry populates the 'Value' held against this Question Index. The elements that make up this function are:

QuestionIndex: A sequential number, commencing at zero to denote the unique question and the order in which the question is presented

Question: Define the question (max 200chars) enclosed in 'Single Quotes'

EditorType: Defines the format of the answer. The options are

TEXT: Open format entry

DATE: Format as your Regional Settings

COMBOBOX: Creates a drop-down list of entries in *Value List*

CALC: This allows you to enter a decimal number. You also have the option to bring up a calculator to calculate the entered value.

CURRENCY: Format as your Regional Settings

SPIN: An Integer only entry allowed. This option also shows arrows from which you can incrementally increase or decrease the displayed Integer

TIME: Format as your Regional Settings

LOOKUP: Used in combination with *LookUpIndex*

If this is defined as blank (two single quotes) then TEXT is assumed.

FullExplanation: A longer explanation that is displayed during data entry

ValueList: If you wish to select entries from a pre-defined list then you should enter the allowable options - separated by a comma. When using a Value List the EditorType must be COMBOBOX. When answering the question a drop-down list is presented showing these options from which a selection can be made. If you are not using this feature then define this with two single quotes

DefaultValue: If you are using a ValueList then you can enter a default Value that prefills the data entry field

LookUpIndex: This allows you to reference any Ostendo table that has a LookUpIndex. (For a list of these refer to section 'Condition Indexes' in Ostendo Help covering 'Reporting'.)

When using a LookUpIndex the EditorType must be LOOKUP. When answering the question a drop-down list is presented showing the current entries in that Table from which a selection can be made.

2. *DataScreenShow*

The format of this function is:

Format: `DataScreenShow:`

This function is used to activate the Data Screen and accept entries. If this is not included in the script then the Data Screen will not be displayed.

3. `DataScreenClose`

The format of this function is:

Format: `DataScreenClose:`

This function is used to close the screen display.

4. `DataScreenSetObjectText`

The format of this function is:

Format: `DataScreenSetObjectText(ObjectID,Value);`

This function is used to populate the text in a defined Object. In the above example it is used in conjunction with the **`DataScreenOnValueEntered`** Procedure. In that Procedure a variable 'Value' has been declared which stores the entered Value against the QuestionIndex number.

This function (which is used against a specific QuestionIndex) takes the content of 'Value' and adds it to the 'Text' for the defined Object ID.

The elements that make up this function are:

ObjectID: The Identifier of the Object in the Graphical Developer

Value: Takes the content of the variable 'Value' to populate the Object

17.1.1.7 Other Functions

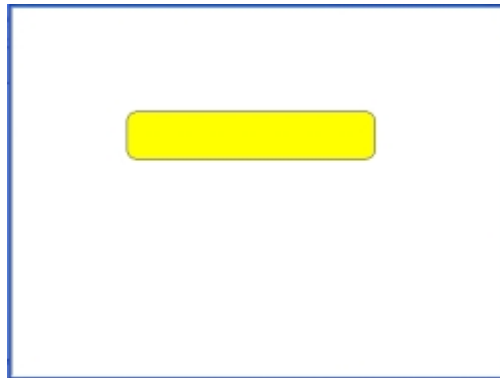
To demonstrate additional functions we should create another Graphical Screen. Therefore go into your Ostendo graphical Developer (OstDesigner.exe under your Ostendo folder) and create a screen as follows:

Click File>New to create a new screen

Click the '**Text Tool**' and add a Text Object to the screen. In the Inspector panel set the object's characteristics as follows:

Alignment:	Centre
Brush:	'Yellow' fill
Font:	Set Font Size as required
Layout:	Centre
Pen:	Pen Style Solid – Black
Roundness	(say) 25
Text:	Remove the Text

The finished result should look something like this



Save the Data Screen

Now create a new Custom Data Screen Script and point it to the above Graphic. Copy and paste the following Script

```
procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);  
begin  
  {This procedure is fired when the control (Ctrl) key is pressed together}  
  {with an alpha key, use it for your own keyboard shortcuts within the script}  
end;  
  
procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);  
begin  
  {This procedure is fired when the enter key is pressed after entering or scanning a value,}  
  {after calling DataScreenQuestion you can process the answer or response here}  
  case QuestionIndex of  
    0:  
      Begin  
        DataScreenSetObjectText(4,Value);  
      End;  
    end;  
  end;  
  
procedure DataScreenObjectClick(ObjectID: Integer);  
begin  
  {This procedure is fired when an object on the top graphical interface  
  is clicked, use the ObjectID to identify the object that was clicked}  
end;  
  
begin  
  DataScreenQuestion(0,'Please Enter a Value','TEXT',' ','');  
  DataScreenShow;  
end.
```

Note: Replace the number in the line `DataScreenSetObjectText(4,Value);` with your Workflow Object ID number. If you run the script it will populate the Object with the data that you entered into the data entry field.

17.1.1.7.1 ProcessBarcode

This takes an entered barcode and interrogates various Tables in Ostendo where barcodes are used and returns a 'string' comprising of the Ostendo Table where the record was found and the SysUniqueID of that record.

However, if the Barcode occurs more than once across Ostendo then a panel will be returned for you to select the correct entry

Format: `ProcessBarcode(Value,True);`

Value: The barcode to be processed

True: The entry options are True or False, 'False' is the default. .If set to True then the Inventory Table is excluded from this process. Note: An Item can be in many locations in Inventory and, if Inventory is included, then a record will be returned for each Location

To prepare for running the script go into Item **100-2000** and enter Barcode **12345**. Also enter barcode **12346** against Item **100-2001**.

Go into the above Data Screen script and replace the second Procedure with the following

```

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
var
  TheCode: String;
begin
  case QuestionIndex of
    0:
      Begin
        TheCode := ProcessBarcode(Value,true);
        DataScreenSetObjectText(4,TheCode);
      End;
    end;
  end;

```

If you run the script and enter Barcode **12346** then the Table (**ITEMMASTER**) and **SysUniqueID** for the Item will be returned to the text for the Object. If you then repeat this for Barcode **12345** then a panel will be presented to show the two variants of the Item. If you select one then that selection will be shown in the Object.

Recap: The final Script would look something like this (Remember to replace the Object ID with yours)

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  case QuestionIndex of
    0:
      Begin
        If Keynumber = 17 then DataScreenClose;
      End;
    end;
  end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
var
  TheCode: String;
begin

```

```

case QuestionIndex of
0:
  Begin
    TheCode := ProcessBarcode(Value,true);
    DataScreenSetObjectText(4,TheCode);
  end;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Enter a Barcode','TEXT',"","");
  DataScreenShow;
end.

```

17.1.1.7.2 DataScreenGetObjectText

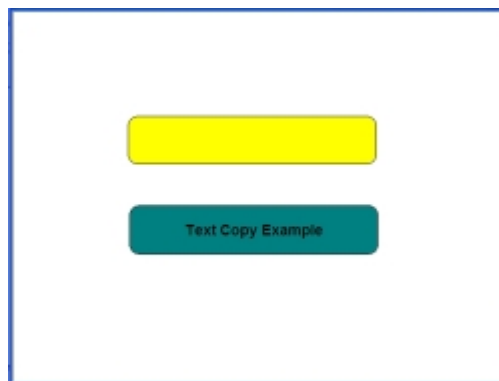
This function is used to get the current content of an Objects 'Text'.

Format: DataScreenGetObjectText(ObjectID);

The element that makes up this function is:

ObjectID: The Identifier of the Object in the Graphical and View Developer. The returned Text from this Object would populate a defined Variable

In this example we will ask if you want to copy the content of first Object and place this in the second Object. Go back into the Graphical Designer and copy the first Object to create a second Object. Note the ID of the Second Object. Enter text 'Text Copy Example' into this object. Your finished screen should look something like this



Go back to the script and replace the script with

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
case QuestionIndex of
0:
  Begin
    If Keynumber = 17 then DataScreenClose;
  End;
end;

```

```

end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
var
  TheText: String;
begin
  case QuestionIndex of
    0:
      Begin
      if Value = 'Yes' then
        begin
          TheText := DataScreenGetObjectText(5);
          DataScreenSetObjectText(4,TheText);
        end
      else
        begin
          DataScreenClose;
        end;
      end;
    end;
  end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Select a value','COMBOBOX','Yes,No','Yes');
  DataScreenShow;
end.

```

You will see in this script that you are asked the Question if you want to copy the Text. If you answer **Yes** then you get the Text of **Object #5** and copy it to **Object #4**. If the answer is **No** then close the screen

17.1.1.7.3 DataScreenObjectLoadPicture

This function is used to load an Object with a picture.

Format: DataScreenObjectLoadPicture(ObjectID,Picture);

The elements that make up this function are:

ObjectID: The Identifier of the Object in the Graphical and View Developer. This MUST be a 'Picture' Object

Picture: The full path of the picture enclosed in single quotes. For example 'C:\Program Files\Ostendo\MyPicture.jpg'

In this example we will ask if you want to add the picture to an Object and if the answer is 'Yes' show the picture.

Go into the Data Screen Script you used above and click on the **'Detail'** tab. Click on the **'Edit'** Button. Add a new Object (must be a picture object) and note the Object ID from the **'Inspector'** panel. **'Save'** the amended form.

Your finished screen should look something like this



Amend the second Procedure to read

```

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  case QuestionIndex of
    0:
      Begin
        if Value = 'Yes' then
          begin
            DataScreenObjectLoadPicture(6,'C:\Program Files\Picture.jpg');
          end
        else
          begin
            DataScreenClose;
          end;
        end;
      end;
    end;
  end;

```

NOTE: Amend the Object ID and 'C:\Program Files\Picture.jpg' to point to the picture on your PC

If you then run the script and press the 'Enter' key you will see that it populates the Picture Object with your selected picture

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  case QuestionIndex of
    0:
      Begin
        If Keynumber = 17 then DataScreenClose;
      End;
    end;
  end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  case QuestionIndex of
    0:
      Begin
        if Value = 'Yes' then

```

```

begin
  DataScreenObjectLoadPicture(6,'C:\Program Files\Picture.jpg');
end
else
begin
  DataScreenClose;
end;
end;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Select a value','COMBOBOX','','Yes,No','Yes');
  DataScreenShow;
end.

```

17.1.1.7.4 DataScreenSaveGraphicalFile

This function will save the current changes by overwriting the source Graphical File. If this is not run then if the Data Screen is amended during the execution of the Script and you then 'close' the changes will not be saved

Format: DataScreenSaveGraphicalFile;

If you run the previous example and close out of the script and then recall it you will see that it has reverted to its original layout

If, however, you add

```
DataScreenSaveGraphicalFile;
```

After

```
DataScreenObjectLoadPicture(6,'C:\Program Files\Picture.jpg');
```

And then re-run it you will find that the amended screen has been retained.

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  case QuestionIndex of
    0:
      Begin
        If Keynumber = 17 then DataScreenClose;
      End;
  end;
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  case QuestionIndex of
    0:
      Begin
        if Value = 'Yes' then

```



```

begin
  DataScreenObjectLoadPicture(6,'C:\Program Files\Picture.jpg');
  DataScreenSaveGraphicalFile;
end
else
begin
  DataScreenClose;
end;
end;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Select a value','COMBOBOX','', 'Yes,No','Yes');
  DataScreenShow;
end.

```

17.1.1.7.5 DataScreenSetEditText

This function will insert data into the input field where it can be amended if required. This could be, for example, a selected line from an Ostendo Table, a hardcoded text, a 'Touch Screen' button, etc. In this example we will use a Ctrl Key to fill the input field and then move that text to a screen Object.

Format: DataScreenSetEditText('Your Text');

The element that makes up this function is:

YourText: The text being set. This can be text entered here (defined in quotes) or can refer to a Var

Go into the above script and replace the complete script with the following:

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
case QuestionIndex of
0:
begin
  If Keynumber = 1 then DataScreenSetEditText('This is my Ctrl-A Text');
end;
end;
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
case QuestionIndex of
0:
begin
  DataScreenSetObjectText(5,Value);
end;
end;
end;

```

```

procedure DataScreenObjectClick(ObjectID: Integer);
begin
  end;

begin
  DataScreenQuestion(0,'Please Enter a Value','TEXT','','');
  DataScreenShow;
end.

```

If you run the Script and press **Ctrl-A** you will see that the Text '**This is my Ctrl-A Text**' will populate the input field. If you press the '**Return**' key on your keyboard then this text will populate the Object in the screen

17.1.1.7.6 DataScreenSetObjectColour

This function allows you to amend the fill colour of an Object within the Graphical Data Screen. The Object must not be currently set to 'Gradient Fill'. The elements that make up this function are:

Format: **DataScreenSetObjectColour(ObjectID, ObjectColour);**

Object ID: The Identifier of the Object in the Graphical Developer

ObjectColour: See the defined colours in the Graphical Editor. The Colour selection is the standard colour preceded with the letters cl. For example you can enter either **clAqua**

In the script used in the previous exercise locate the line
 DataScreenSetObjectText(5,Value);

And insert this immediately after it
 DataScreenSetObjectColour(5,clAqua);

When you run the script it will, in addition to inserting the Text into Object ID **5** change the colour of the Object to **Aqua**

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  case QuestionIndex of
    0:
      begin
        If Keynumber = 1 then DataScreenSetEditText('This is my Ctrl-A Text');
      end;
      end;
  end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  case QuestionIndex of
    0:
      Begin
        DataScreenSetObjectText(5,Value);
        DataScreenSetObjectColour(5,clAqua);
      End;
  end;

```

```

end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Enter a Value','TEXT',"","");
  DataScreenShow;
end.

```

17.1.1.7.7 DataScreenSetObjectGradientColour

This function allows you to amend the Gradient fill colour of an Object within the Graphical Data Screen. The Object must be currently set to 'Gradient Fill'. The elements that make up this function are:

Format: **DataScreenSetObjectGradientColour(ObjectID, BeginColour,EndColour);**

Object ID: The Identifier of the Object in the Graphical Developer

BeginColour: The first colour in the Gradient. See the defined colours in the Graphical Editor. The Colour selection is the standard colour preceded with the letters cl. For example you can enter either **clAqua**

EndColour: The second colour in the Gradient. See the defined colours in the Graphical Editor.

Go into the Data Screen Script you used in previous exercise and click on the '**Detail**' tab. Click on the '**Edit**' Button. Locate the second Object and change its 'Brush' to 'Gradient Fill'. Save the change then return to the script.

In the script replace the line

```
DataScreenSetObjectColour(5,claqua);
```

With

```
DataScreenSetObjectGradientColour(5,claqua,clblue);
```

When you run the script it will, in addition to inserting the Text into Object ID **5** change the Gradient Fill colour of the Object

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
case QuestionIndex of
  0:
    begin
      If Keynumber = 1 then DataScreenSetEditText('This is my Ctrl-A Text');
    end;
  end;
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
case QuestionIndex of
  0:
    Begin

```

```

    DataScreenSetObjectText(5,Value);
    DataScreenSetObjectGradientColour(5,claqua,clblue);
End;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
    DataScreenQuestion(0,'Please Enter a Value','TEXT',"","");
    DataScreenShow;
end.

```

17.1.1.7.8 DataScreenSetObjectHint

This function allows you to add or amend a Hint to an Object. A 'Hint' appears whenever the cursor is passed over the object. The elements that make up this function are:

Format: **DataScreenSetObjectHint(ObjectID,Hint);**

Object ID: The Identifier of the Object in the Graphical Developer

Hint: The hint that will replace the current hint.

In the script used in the previous exercise replace

```
DataScreenSetObjectGradientColour(5,claqua,clblue);
```

With

```
DataScreenSetObjectHint(5,'This is the Hint against this Object');
```

If you run the script and move the mouse over the Object then no hint will be present. However the Hint is activated by pressing the 'Return' key when answering QuestionIndex 0 Therefore press the return Key and - once again - move the cursor over the Object to view the Hint

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
case QuestionIndex of
    0:
        begin
            If Keynumber = 1 then DataScreenSetEditText('This is my Ctrl-A Text');
        end;
        end;
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
case QuestionIndex of
    0:
        Begin
            DataScreenSetObjectText(5,Value);
            DataScreenSetObjectHint(5,'This is the Hint against Object 5');

        End;

```

```

    end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Enter a Value','TEXT',",",",");
  DataScreenShow;
end.

```

17.1.1.7.9 DataScreenSetObjectTransparency

This function allows you to define the 'opaqueness' of the Object. The elements that make up this function are:

Format: DataScreenSetObjectTransparency(ObjectID,Transparency);

Object ID: The Identifier of the Object in the Graphical Developer

Transparency: The amount of 'opaqueness'. This ranges from 0 = solid to 100 = Invisible

Go into the **'Detail'** tab and click on the **'Edit'** Button. Click on the second Object and drag it until it partially covers the first Object.

In the script replace

```
DataScreenSetObjectText(5,Value);
```

With

```
DataScreenSetObjectTransparency(5,60);
```

If you run the script and press the return Key you will see that the Transparency of the Object has changed

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  case QuestionIndex of
    0:
      begin
        DataScreenSetObjectTransparency(5,60);
      end;
  end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0,'Please Enter a Value','TEXT',",",",");

```

```
DataScreenShow;  
end.
```

17.1.1.7.10 DataScreenSetObjectVisible

This function allows you to turn off or on the visibility of an Object. The elements that make up this function are:

Format: DataScreenSetObjectVisible(ObjectID,Value);

Object ID: The Identifier of the Object in the Graphical Developer

Value: This can be True or False

In the script used in the previous exercise replace the second Procedure with the following:

```
procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);  
begin  
  case QuestionIndex of  
    0:  
      Begin  
      if Value = 'Yes' then  
        begin  
          DataScreenSetObjectVisible(5,True);  
        end  
      else  
        begin  
          DataScreenSetObjectVisible(5,False);  
        end;  
      end;  
      end;  
      end;  
end;
```

If you run the script and select 'Yes' you will see that the Object is visible. If you now select 'No' it will disappear.

Recap: The final Script would look something like this

```
procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);  
begin  
  case QuestionIndex of  
    0:  
      Begin  
      If Keynumber = 17 then DataScreenClose;  
      End;  
    end;  
end;
```

```
procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);  
begin  
  case QuestionIndex of  
    0:  
      Begin  
      if Value = 'Yes' then  
        begin  
          DataScreenSetObjectVisible(5,True);  
        end  
      end
```

```

else
begin
DataScreenSetObjectVisible(5,False);
end;
end;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
DataScreenQuestion(0,'Please Select a value','COMBOBOX','Yes,No','Yes');
DataScreenShow;
end.

```

17.1.1.7.11 DataScreenChangeScheme

This function allows you to go to another Scheme in the same Ostendo Graphic. You should first create the second Scheme. To do this go into the Graphic (you can access this via the 'Edit' button in the 'Detail' tab of the Data Screen Script). You should note the name of the current Scheme on the Layout Toolbar. Click on the 'New Scheme' Icon and generate a new Scheme. Note the ID of that Scheme. You should now have a single Graphic with two schemes. 'Save' the graphic.

The elements that make up this function are:

Format: DataScreenChangeScheme(SchemeID);

SchemeID: The Identifier of the Scheme which you are changing to

Amend the script in the previous exercise such that the second procedure looks like this.

```

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
case QuestionIndex of
0:
Begin
if Value = 'Yes' then
begin
DataScreenChangeScheme('Scheme2');
end
else
begin
DataScreenChangeScheme('Scheme1');
end;
end;
end;
end;

```

**** Where **Scheme1** and **Scheme2** are the Scheme Identities

If you now run the Data Screen Script you will see that it will alternate between the two schemes depending on the entry made.

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  case QuestionIndex of
    0:
      Begin
        If Keynumber = 17 then DataScreenClose;
      End;
    end;
  end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
begin
  case QuestionIndex of
    0:
      Begin
        if Value = 'Yes' then
          begin
            DataScreenChangeScheme('Scheme2');
          end
        else
          begin
            DataScreenChangeScheme('Scheme1');
          end;
        end;
      end;
    end;
  end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
end;

begin
  DataScreenQuestion(0, 'Please Select a value', 'COMBOBOX', 'Yes,No', 'Yes');
  DataScreenShow;
end.

```

17.1.1.7.12 DataScreenActiveScheme

This function allows you to determine the Identity of the current Active Scheme

The elements that make up this function are:

Format: DataScreenActiveScheme(SchemeID);

SchemeID: The Identifier of the Scheme which you are changing to

Amend the script in the previous exercise such that the second procedure looks like this.

```

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
Var
  ActiveScheme: String;
begin
  case QuestionIndex of
    0:

```



```

Begin
if Value = 'Yes' then
  begin
    DataScreenChangeScheme('Scheme2');
    ActiveScheme := DataScreenActiveScheme;
    Showmessage(ActiveScheme);
  end
else
  begin
    DataScreenChangeScheme('Scheme1');
    ActiveScheme := DataScreenActiveScheme;
    Showmessage(ActiveScheme);
  end;
end;
end;
end;

```

If you now run the Data Screen Script you will see that it will alternate between the two schemes depending on the entry made and return a Showmessage telling you which Scheme you are currently on.

Recap: The final Script would look something like this

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  case QuestionIndex of
    0:
      Begin
        If Keynumber = 17 then DataScreenClose;
      End;
    end;
  end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
Var
ActiveScheme: String;
begin
  case QuestionIndex of
    0:
      Begin
        if Value = 'Yes' then
          begin
            DataScreenChangeScheme('Scheme2');
            ActiveScheme := DataScreenActiveScheme;
            Showmessage(ActiveScheme);
          end
        else
          begin
            DataScreenChangeScheme('Scheme1');
            ActiveScheme := DataScreenActiveScheme;
            Showmessage(ActiveScheme);
          end;
        end;
      end;
    end;
  end;

```

```

procedure DataScreenObjectClick(ObjectID: Integer);
begin
  end;

begin
  DataScreenQuestion(0,'Please Select a value','COMBOBOX','','Yes,No','Yes');
  DataScreenShow;
end.

```

17.1.2 Simple Example

In this exercise we will create an Inquiry Screen showing specific fields from a selected Item Code. This uses other Ostendo's Scripting functions to get the data from the ITEMMASTER table and populate an Object with that data

Firstly create the Graphic containing Objects covering the following fields.

- Item Code
- Description
- Item Unit
- Std Sell Price
- Std Buy Price
- On Hand Qty

Take a note of the ID of each of the 6 main Objects

In Ostendo go into **File>Custom Scripts** and create a Script called (say) **'ItemInquiry'**. In the **'Detail'** tab select Style **'Custom Data Screen'**. In the **'Data Screen File'** field point it to the Data Screen you created above.

Click on the 'Script' tab and enter the following script

```

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  {This procedure is fired when the control (Ctrl) key is pressed together}
  {with an alpha key, use it for your own keyboard shortcuts within the script}
case QuestionIndex of
  0:
    Begin
      If Keynumber = 17 then DataScreenClose;
    end;
  end;
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
var
  TheSellPrice, TheBuyPrice: double;
begin
  {This procedure is fired when the enter key is pressed after entering or scanning a value,}
  {after calling DataScreenQuestion you can process the answer or response here}
case QuestionIndex of
  0:
    begin
      DataScreenSetObjectText(5,Value);
      DataScreenSetObjectText(6,GetStringFromTable('ITEMMASTER',
        'ITEMDESCRIPTION', 'ITEMCODE', Value));
    end;
  end;

```

```

DataScreenSetObjectText(11,GetStringFromTable('ITEMMASTER', 'ITEMUNIT',
'ITEMCODE', Value));
TheSellPrice := GetDoubleFromTable('ITEMMASTER', 'STDSELLPRICE',
'ITEMCODE', Value);
DataScreenSetObjectText(12,FormatFloat('$###,##0.00',TheSellPrice));
TheBuyPrice := GetDoubleFromTable('ITEMMASTER', 'STDBUYPRICE',
'ITEMCODE', Value);
DataScreenSetObjectText(15,FormatFloat('$###,##0.00',TheBuyPrice));
DataScreenSetObjectText(16,GetStringFromTable('ITEMMASTER',
'ONHANDQTY', 'ITEMCODE', Value));
DataScreenQuestion(0,'Please Select the Item Code','LOOKUP','Select another
Item Code or Ctrl-Q to Quit',"",1004);
end;
end;
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin
  {This procedure is fired when an object on the top graphical interface
  is clicked, use the ObjectID to identify the object that was clicked}
end;

begin
  DataScreenQuestion(0,'Please Select the Item Code','LOOKUP','Select the Item
  Code and click the Enter Key',"",1004);
  DataScreenShow;
end.

```

Points to Note:

1. Against each DataScreenSetObjectText function amend the Object ID reference number to suit your Graphic
2. The Price fields have been converted to decimal using the FormatFloat instruction. You will need to amend the '\$' symbol if your base currency symbol is different.

Having created the Item Inquiry Screen go to File>Custom Data Screens and click on the Data Screen you created above to run that screen.

17.1.3 Advanced Example

This example lets you select Items and quantities to be transferred to another Location after which the Script will generate a Warehouse Transfer batch that you can see in Ostendo under **Inventory>Inventory Transfer**

```

Const
TheFinishChar = 'Z';
TheDeleteChar = 'D';
TheQuitChar = 'Q';
ScanMessage = 'Please scan or enter the barcode or (Enter Z) to Finish and create the
Transfer' + chr(13) +
'or (Enter Q) to Quit and cancel the Transfer' + chr(13) +
'or (Enter D) to Delete a sepecific transfer line';

var
TheMemoText,TheMemoText1,TheMemoText2: string;
TheTransferList: TStringlist;
Mappings: TStringlist;

```

```
LocationList: TStringList;  
TheBatchID,TheLocations,TheFromLocation,TheToLocation,TheTitle: string;  
QuitNoPost: Boolean;
```

```
procedure CreateLocationList;  
var  
x: integer;  
begin  
  {The locations are added here}  
  {The first field is the shortcode, the second is the Actual Warehouse  
  and the third is the actual location}  
  LocationList.add('Main,Main,Primary');  
  LocationList.add('Van01,Secondary,Bulk');  
  
  for x := 0 to (LocationList.count-1) do  
  begin  
    if x = 0 then  
    begin  
      TheLocations := parsestring(LocationList.strings[x],',',0);  
    end  
    else  
    begin  
      TheLocations := TheLocations + ',' + parsestring(LocationList.strings[x],',',0);  
    end;  
  end;  
end;
```

```
function GetActualWarehouse(PassedShortCode: string):string;  
var x: integer;  
begin  
  for x := 0 to (LocationList.count - 1) do  
  begin  
    if parsestring(LocationList.strings[x],',',0) = PassedShortCode then  
    begin  
      result := parsestring(LocationList.strings[x],',',1);  
    end;  
  end;  
end;
```

```
function GetActualLocation(PassedShortCode: string):string;  
var x: integer;  
begin  
  for x := 0 to (LocationList.count - 1) do  
  begin  
    if parsestring(LocationList.strings[x],',',0) = PassedShortCode then  
    begin  
      result := parsestring(LocationList.strings[x],',',2);  
    end;  
  end;  
end;
```

```
function IsLocationValid(PassedShortCode: string):string;  
var  
x: integer;  
begin  
  result := 'No';
```

```

for x := 0 to (LocationList.count - 1) do
begin
  if (parsestring(LocationList.strings[x],',',0) = PassedShortCode) then
    begin
      result := 'Yes';
    end;
  end;
end;

procedure CreateTransferBatch;
begin
  TheBatchID := GetSQLResult('select gen_id(INVENTORYTRANSFERNO,1) from
RDB$DATABASE');
  Mappings.add('TRANSFERNO=' + TheBatchID);
  Mappings.add('TRANSFERREFERENCE=Data Collection Batch');
  Mappings.add('TRANSFERSTATUS=InProgress');
  Mappings.add('TRANSFERDATE=' + datetostr(date));
  Mappings.add('TRANSFERNOTES=Created by the Data Collection Module');
  InsertRecord('INVENTORYTRANSFERS',Mappings.text);
  CreateTransferLines;
end;

procedure CreateTransferLines;
var
  x: integer;
begin
for x := 0 to (TheTransferList.count - 1) do
  begin
    if ((parsestring(TheTransferList.strings[x],',',1) = 'A') and
      (parsestring(TheTransferList.strings[x],',',4) <> '0')) then
      begin
        Mappings.clear;
        Mappings.add('TRANSFERNO=' + TheBatchID);
        Mappings.add('ITEMCODE=' + parsestring(TheTransferList.strings[x],',',2));
        Mappings.add('UNIT=' + parsestring(TheTransferList.strings[x],',',3));
        Mappings.add('FROMWAREHOUSE=' + GetActualWarehouse(TheFromLocation));
        Mappings.add('TOWAREHOUSE=' + GetActualWarehouse(TheToLocation));
        Mappings.add('FROMLOCATION=' + GetActualLocation(TheFromLocation));
        Mappings.add('TOLOCATION=' + GetActualLocation(TheToLocation));
        Mappings.add('TRANSFERQTY=' + parsestring(TheTransferList.strings[x],',',4));
        Mappings.add('INVENTORYONHANDQTY=' +
parsestring(TheTransferList.strings[x],',',5));
        InsertRecord('INVENTORYTRANSLINES',Mappings.text);
      end;
    end;
  end;
end;

procedure DataScreenOnCtrlKey(QuestionIndex: Integer; KeyNumber: Integer);
begin
  {This procedure is fired when the control (Ctrl) key is pressed together}
  {with an alpha key, use it for your own keyboard shortcuts within the script}
end;

procedure DataScreenOnValueEntered(QuestionIndex: Integer; Value: String);
var

```

TheR: boolean;

TheCode,TheTable,TheID,TheDescription,TheQty,TheLineNoStr,TheUnit,TheQOH,TheCount: **string**;

TheNewLineNumber: **string**;

TheLineNumber,Z: integer;

begin

{This procedure is fired when the enter key is pressed after entering or scanning a value,}
{after calling DataScreenQuestion you can process the answer or response here}

case QuestionIndex **of**

0:

begin

TheFromLocation := Value;

if (IsLocationValid(TheFromLocation) = 'Yes') **then**

begin

DataScreenSetObjectText(101, Value);

DataScreenQuestion(1,'To Warehouse-Location','COMBOBOX','Select the To Location',TheLocations,");

end

else

begin

DataScreenQuestion(0,'From Warehouse-Location','COMBOBOX','Select the From Location',TheLocations,");

end;

end;

1:

begin

TheToLocation := Value;

if (IsLocationValid(TheToLocation) = 'Yes') **then**

begin

DataScreenSetObjectText(102, Value);

DataScreenQuestion(2,'Scan Barcode','TEXT',ScanMessage,");

end

else

begin

DataScreenQuestion(1,'To Warehouse-Location','COMBOBOX','Select the To Location',TheLocations,");

end;

end;

2:

begin

if uppercase(Value) = TheFinishChar **then**

begin

QuitNoPost := False;

DataScreenClose;

end;

if uppercase(Value) = TheQuitChar **then**

begin

QuitNoPost := true;

DataScreenClose;

end;

if uppercase(Value) = TheDeleteChar **then**

begin

DataScreenQuestion(4,'Enter Deletion Line','SPIN','Please enter the line number to be Deleted',");

end

```

else
begin
  TheCode := ProcessBarcode(Value, true);
  if ((TheCode = "") or (TheLineNumber > 39)) then
  begin
    if (TheLineNumber > 39) then
    begin
      showmessage('You reached the limit for a single transfer - please post this transfer
and create a new one for any remaining items');
    end;
    DataScreenQuestion(2,'Scan Barcode','TEXT',ScanMessage,"");
  end
  else
  begin
    TheTable := ParseString(TheCode, '=',0);
    TheID := ParseString(TheCode, '=',1);
    if ((TheTable <> 'ITEMMASTER') and (TheTable <> 'ITEMBARCODES') and
(TheTable <> 'ITEMUNITS')) then
    begin
      DataScreenQuestion(2,'Scan Barcode','TEXT',ScanMessage,"");
      exit;
    end;

    if ((TheTable = 'ITEMMASTER') or (TheTable = 'ITEMUNITS')) then
    begin
      TheCount := GetSQLResult('select count(sysuniqueid) from Itemmaster where
sysuniqueid = ' + TheID);
      if (TheCount <> '0') then
      begin
        TheCode := GetSQLResult('select Itemcode from Itemmaster where sysuniqueid =
' + TheID);
        TheUnit := GetSQLResult('select Itemunit from Itemmaster where sysuniqueid = ' +
TheID);
      end
      else
      begin
        TheCode := GetSQLResult('select Itemcode from Itemunits where sysuniqueid = '
+ TheID);
        TheUnit := GetSQLResult('select tounit from Itemunits where sysuniqueid = ' +
TheID);
      end;
      TheDescription := GetSQLResult('select Itemdescription from Itemmaster where
itemcode = ' + TheCode + '');
      TheCount := GetSQLResult('select count(sysuniqueid) from Inventory where
itemcode = ' + TheCode + ' and Inventoryunit = ' + TheUnit + '');
      if (TheCount <> '0') then
      begin
        TheQOH := GetSQLResult('select sum(inventoryqty) from Inventory where
itemcode = ' + TheCode + ' and Inventoryunit = ' + TheUnit + '');
      end
      else
      begin
        TheQOH := '0';
      end;
    end;
  end
  if (TheTable = 'ITEMBARCODES') then

```

```

    begin
        TheCode := GetSQLResult('select Itemcode from Itembarcodes where sysuniqueid
= ' + TheID);
        TheDescription := GetSQLResult('select Itemdescription from Itemmaster where
itemcode = ' + TheCode + '');
        TheUnit := GetSQLResult('select Itemunit from Itembarcodes where sysuniqueid = '
+ TheID);
        TheCount := GetSQLResult('select count(sysuniqueid) from Inventory where
itemcode = ' + TheCode + ' and Inventoryunit = ' + TheUnit + '');
        if (TheCount <> '0') then
            begin
                TheQOH := GetSQLResult('select sum(inventoryqty) from Inventory where
itemcode = ' + TheCode + ' and Inventoryunit = ' + TheUnit + '');
            end
            else
                begin
                    TheQOH := '0';
                end;
            end;

        DataScreenQuestion(3,'Enter Qty','CALC','Please enter the Qty to transfer for Item: '
+ TheCode + '(' + TheDescription + ')' + chr(13) + 'If 0 is entered then the current Item will
not be transferred',",");
        end;
    end;
end;
3:
begin
    TheQty := Value;
    TheLineNumber := TheLineNumber + 1;
    if TheLineNumber < 10 then
        begin
            TheLineNoStr := '0' + intostr(TheLineNumber);
        end
        else
            begin
                TheLineNoStr := intostr(TheLineNumber);
            end;
    if (TheLineNumber = 21) then
        begin
            TheMemoText := "";
        end;

    if TheMemoText = " then
        begin
            TheMemoText := '[' + TheLineNoStr + '] Item: ' + TheCode + '(' + TheUnit + ') Qty: ' +
TheQty + '(' + TheDescription + ')';
            TheTransferList.add(TheLineNoStr + ',A,' + TheCode + ',' + TheUnit + ',' + TheQty + ','
+ TheQOH + ',' + TheDescription);
        end
        else
            begin
                TheMemoText := TheMemoText + chr(13) + '[' + TheLineNoStr + '] Item: ' + TheCode
+ '(' + TheUnit + ') Qty: ' + TheQty + '(' + TheDescription + ')';
                TheTransferList.add(TheLineNoStr + ',A,' + TheCode + ',' + TheUnit + ',' + TheQty + ','
+ TheQOH + ',' + TheDescription);
            end;
        end;
end;

```



```

end;

if (TheLineNumber < 21) then
begin
  DataScreenSetObjectText(10003, TheMemoText);
end
else
begin
  DataScreenSetObjectText(10004, TheMemoText);
end;

DataScreenQuestion(2,'Scan Barcode','TEXT',ScanMessage,"");
end;
4:
begin
TheMemoText2 := "";
TheTransferList.strings[strtoint(value)-1] := '[' + intostr(strtoint(value) + 1) + ',Deleted';
for z := 0 to (TheTransferList.count-1) do
begin
  if (z < 9) then
  begin
    TheNewLineNumber := '0' + intostr(z + 1);
  end
  else
  begin
    TheNewLineNumber := intostr(z + 1);
  end;
end;

if (parsestring(TheTransferList.strings[z],',',1) = 'A') then
begin
  if ((z = 0) or (z = 20)) then
  begin
    if (z = 0) then
    begin
      TheMemoText1 := '[' + TheNewLineNumber + '] Item: ' +
        parsestring(TheTransferList.strings[z],',',2) + '(' +
        parsestring(TheTransferList.strings[z],',',3) + ') Qty: ' +
        parsestring(TheTransferList.strings[z],',',4) + '(' +
        parsestring(TheTransferList.strings[z],',',6) + ')';
    end
    else
    begin
      TheMemoText2 := '[' + TheNewLineNumber + '] Item: ' +
        parsestring(TheTransferList.strings[z],',',2) + '(' +
        parsestring(TheTransferList.strings[z],',',3) + ') Qty: ' +
        parsestring(TheTransferList.strings[z],',',4) + '(' +
        parsestring(TheTransferList.strings[z],',',6) + ')';
    end;
  end
  end
  else
  begin
    if (z < 20) then
    begin
      TheMemoText1 := TheMemoText1 + chr(13) + '[' + TheNewLineNumber + '] Item:
+
      parsestring(TheTransferList.strings[z],',',2) + '(' +

```

```

        parsestring(TheTransferList.strings[z],',',3) + ') Qty: ' +
        parsestring(TheTransferList.strings[z],',',4) + ' (' +
        parsestring(TheTransferList.strings[z],',',6) + ');
    end
  else
    begin
      TheMemoText2 := TheMemoText2 + chr(13) + '[' + TheNewLineNumber + '] Item:
' +
        parsestring(TheTransferList.strings[z],',',2) + '(' +
        parsestring(TheTransferList.strings[z],',',3) + ') Qty: ' +
        parsestring(TheTransferList.strings[z],',',4) + ' (' +
        parsestring(TheTransferList.strings[z],',',6) + ');

    end;
  end;
end
else
  begin
    if (z < 20) then
      begin
        if z = 0 then
          begin
            TheMemoText1 := '[' + TheNewLineNumber + '] Deleted';
          end
        else
          begin
            TheMemoText1 := TheMemoText1 + chr(13) + '[' + TheNewLineNumber + ']
Deleted';
          end;
        end
      else
        begin
          if (z = 20) then
            begin
              TheMemoText2 := '[' + TheNewLineNumber + '] Deleted';
            end
          else
            begin
              TheMemoText2 := TheMemoText2 + chr(13) + '[' + TheNewLineNumber + ']
Deleted';
            end;
          end;
        end;
      end;
    DataScreenSetObjectText(10003, TheMemoText1);
    DataScreenSetObjectText(10004, TheMemoText2);
    if (z < 20) then
      begin
        TheMemoText := TheMemoText1;
      end
    else
      begin
        TheMemoText := TheMemoText2;
      end;
    DataScreenQuestion(2,'Scan Barcode','TEXT',ScanMessage,"");
  end;
end;

```

```

    end; {Case}
end;

procedure DataScreenObjectClick(ObjectID: Integer);
begin

end;

var
    x: Integer;
    TheCompanyName: string;
begin
try
    LocationList := TStringList.create;
    TheTransferList := TStringList.Create;
    Mappings := TStringList.Create;
    TheMemoText := "";
    QuitNoPost := True;
    CreateLocationList;
    DataScreenQuestion(0,'From Warehouse-Location','COMBOBOX','Select the From
Location',TheLocations,"");
    DataScreenShow;
    if ((TheTransferList.count > 0) and (QuitNoPost = False)) then
    begin
        CreateTransferBatch;
        showmessage('Transfer: ' + TheBatchID + ' Created');
    end
    else
    begin
        showmessage('No Transfer Created');
    end;
    finally
        LocationList.free;
        TheTransferList.Free;
        Mappings.free;
    end;
end.

```

17.2 Custom Data Entry

Custom Data Entry allows you to create your own Data Entry Grid into which you can enter multiple lines, and subsequently action these records; usually by updating Ostendo

17.2.1 Data Entry Form

Let us start by creating the Form itself and then look at each field type that can be included in the Data Entry Form.

In Ostendo go into **File>Custom Scripts** and create a 'Standard' Script called (say) '**DataEntryScreen**'.

For the opening exercise we will create the Data Entry Form and, later, show the different field formats that can be used.

17.2.2 Creating the Input Panel

Enter the following into the 'Script' tab and then 'run' the script.

```
begin  
  {create the data entry form}  
  DataEntryCreate('Script Data Entry',500,720);  
  DataEntryShow;  
end.
```

This script contains two Data Entry Functions. These are

1.1.1. DataEntryCreate

The first function will create the Data Entry panel. The elements that make up this function are:

Format: DataEntryCreate(Title{Opt Default='Data Entry'}, FormHeight{Opt Default=400}, FormWidth{Opt Default=550});;

Where the elements represent the following

Title: The Title that will appear at the top of the form

Height: The Height of the Form in Pixels

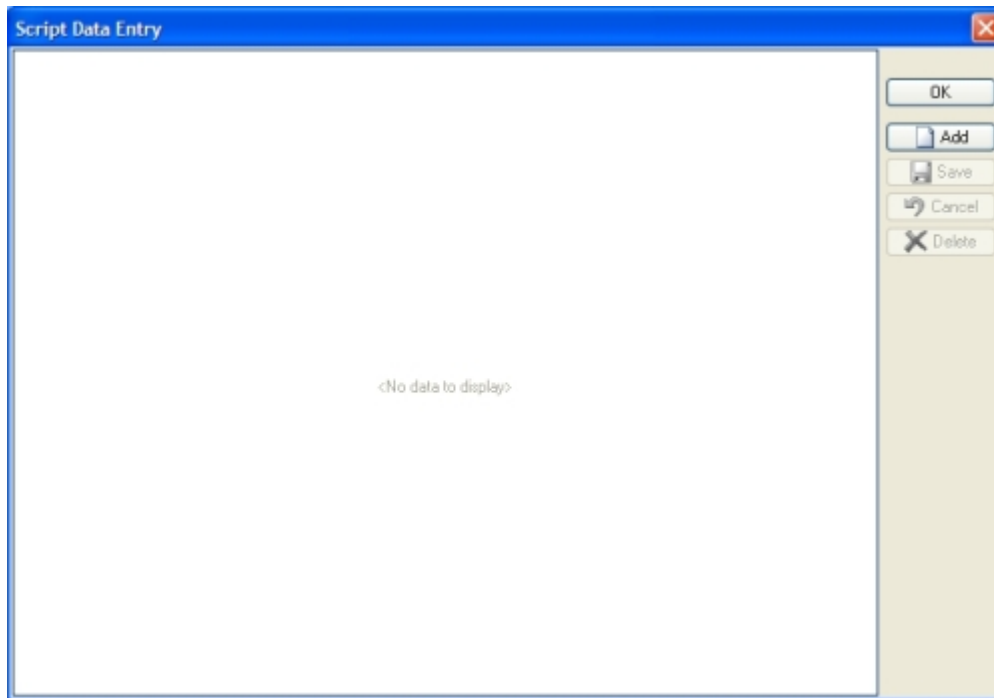
Width: The Width of the Form in Pixels

1.1.2.DataEntryShow

The second function simply instructs Ostendo to display the Entry Form

Format: DataEntryShow;

If you click the 'Run' button and the following (Blank) Data Entry Form will be presented



You can see that the Form contains standard Add, Save, Cancel, Delete, and OK Buttons. What it does not contain are the fields themselves.

17.2.3 Defining the fields

Replace the script you created above with the following

```
begin
  {create the data entry form}
  DataEntryCreate('Script Data Entry',500,720);
  {Create columns editor types shown below}
  DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
  DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
  DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
  DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
  DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
  DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
  DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}
  DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
  DataEntryShow;
end.
```

If you run the script you can click on the 'Add' button to create a new record. This script contains a function called

DataEntryColumnCreate

The format of this function is:

Format: DataEntryColumnCreate(Caption, ColumnWidth{Opt Default=100}, EditorType{Opt Default='TEXT'}, LookupIndex{Opt Default=0});

It is used to define the sequence and format of the data entry columns. The elements that make up this function are:

Caption: The caption that populates the column heading

ColumnWidth: The width of the column

EditorType: Defines the format of the entry field. The options are

TEXT: Open format entry

DATE: Entry format as your Regional Settings

CALC: This allows you to enter a decimal number. You also have the option to bring up a calculator to calculate the entered value.

CURRENCY: Format as your Regional Settings

SPIN: An Integer only entry allowed. This option also shows arrows from which you can incrementally increase or decrease the displayed Integer

TIME: Entry format as your Regional Settings

CHECKBOX: A 'Yes/No 'Boolean' checkbox

LOOKUP: Used in combination with **LookUpIndex**

LookUpIndex: This allows you to reference any Ostendo table that has a LookUpIndex.

(For a list of these refer to section 'Condition Indexes' in Ostendo Help covering 'Reporting'.)

When using a LookUpIndex the EditorType must be LOOKUP. When answering the question a drop-down list is presented showing the current entries in that Table from which a selection can be made.

17.2.4 Other Data Entry Functions

Let's have a look at some of the other functions for use with Screen Data Scripts

17.2.4.1 DataEntrySetLabel

This function enables you to create a Label that appears across the top of the grid. The format of this function is:

Format: **DataEntrySetLabel(Title);**

The elements that make up this function are:

Title: The Title of the Grid

To see this in action extend the Script you created above to include one extra line just above the DataEntryShow Function. For example

```
DataEntrySetLabel('This is the Test Entry Grid');
```

The Script should now look like this:

```
begin
  {create the data entry form}
  DataEntryCreate('Script Data Entry',500,720);
  {Create columns editor types shown below}
  DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
  DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
  DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
  DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
  DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
  DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
  DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}
```

```
DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
DataEntrySetLabel('This is the Test Entry Grid');
DataEntryShow;
end.
```

If you now run the script the Label will be presented at the top of the Grid.

17.2.4.2 DataEntryNewRecordValuesSet

This function enables you to prefill fields when creating records. The format of this function is:

Format: DataEntryNewRecordValuesSet(RecordValues);

The elements that make up this function are:

RecordValues: The default values in each field separated by a comma

To see this in action extend the Script you created above to include this extra line just above the DataEntryShow Function. For example

```
DataEntryNewRecordValuesSet(
'100-2000,Description,27/01/2009,4.25,10:00,20.12,True,5')
```

The Script should now look like this:

```
begin
{create the data entry form}
DataEntryCreate('Script Data Entry',500,720);
{Create columns editor types shown below}
DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
DataEntryColumnCreate('Date Editor',80,'DATE'); {2}
DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
DataEntryColumnCreate('Check Editor',80,'CHECKBOX'); {6}
DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
DataEntrySetLabel('This is the Test Entry Grid');
DataEntryNewRecordValuesSet(
'100-2000,Description,27/01/2009,4.25,10:00,20.12,True,5');
DataEntryShow;
end.
```

If you now run the script and click the 'Add' button to create a new record you will find that the fields have been pre-filled with these default values.

17.2.4.3 DataEntryCellValueGet

This is used to get the value from a specific field within a record in the Table

Format: DataEntryCellValueGet(RecordIndex, ColumnIndex);

The elements that make up this function are:

RecordIndex: The record number being addressed (starting from 0)

ColumnIndex: The field number in the record (starting from 0)

To see this in action copy the following script and overwrite your existing script. We have added the Variable x to represent the RecordIndex and enable us to look at the 4th field (ColumnIndex 3)

in each record

```

Var
x: Integer;
begin
  {create the data entry form}
  DataEntryCreate('Script Data Entry',500,720);
  {Create columns editor types shown below}
  DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
  DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
  DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
  DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
  DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
  DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
  DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}
  DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
  DataEntrySetLabel('This is the Test Entry Grid');
  DataEntryNewRecordValuesSet(
'100-2000,Description,27/01/2009,4.25,10:00,20.12,True,5');
  DataEntryShow;
  for x := 0 to DataEntryRecordCount - 1 do
    begin
      showmessage('The Cell Value is ' + DataEntryCellValueGet(x, 3));
    end;
  end.

```

Having copied the Script you should now:

- Run the script
- Click the '**Add**' button to create the first record.
- Go to the '**Calc Editor**' field and change the Quantity.
- 'Save' the record
- Click the '**Add**' button to create a second record.
- Go to the '**Calc Editor**' field and change the Quantity.
- '**Save**' the record
- Finally, Click to '**OK**' button to view the content of the filed in each record.

17.2.4.4 DataEntryCellValueSet

This function enables you to populate a specific field within a record with a defined value. The format of this function is:

Format: **DataEntryCellValueSet(RecordIndex,ColumnIndex,CellValue);**

The elements that make up this function are:

RecordIndex: The record number being addressed (starting from 0)

ColumnIndex: The field number in the record (starting from 0)

CellValue: The value to populate the field

To see this in action use the following script

```

begin
  {create the data entry form}
  DataEntryCreate('Script Data Entry',500,720);
  {Create columns editor types shown below}
  DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
  DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}

```



```

DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}
DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
DataEntrySetLabel('This is the Test Entry Grid');
DataEntryCellValueSet(0,1,'The Description');
DataEntryShow;
end.

```

You will see that, for the first record only, the 'Text Editor' field will be pre-populated with the above value

17.2.4.5 DataEntryColumnCount

This function enables you to determine the number of columns in the Grid. The format of this function is:

Format: DataEntryColumnCount;

To demonstrate this add this Showmessage line immediately after the DataEntryShow function of your script. I.e.

```
Showmessage('The Number of Columns is ' + IntToStr(DataEntryColumnCount));
```

Your resultant script should look like this

```

begin
{create the data entry form}
DataEntryCreate('Script Data Entry',500,720);
{Create columns editor types shown below}
DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}
DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
DataEntrySetLabel('This is the Test Entry Grid');
DataEntryCellValueSet(0,1,'The Description');
DataEntryShow;
Showmessage('The Number of Columns is ' + IntToStr(DataEntryColumnCount));
end.

```

If you run the script and click the OK button the message will be returned with the number of columns.

17.2.4.6 DataEntryRecordCount

When you enter the records they are stored in a temporary table until you tell Ostendo what to do with them. This function simply counts the number of records in this temporary table. The format of this function is:

Format: DataEntryRecordCount;

Using the following Script we will access this temporary table and return, after the OK button has been clicked, the number of records entered.

```

Var
  x: Integer;
begin
  {create the data entry form}
  DataEntryCreate('Script Data Entry',500,720);
  {Create columns editor types shown below}
  DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
  DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
  DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
  DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}
  DataEntryColumnCreate('Time Editor',80,'TIME'); {4}
  DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}
  DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}
  DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}
  DataEntryShow;
  DataEntryRecordCount;
  Showmessage('The number of records entered is ' + IntToStr(DataEntryRecordCount));
end.

```

17.2.5 Data Entry Procedures

The following Procedures have been developed specifically for Custom Data Entry

17.2.5.1 DataEntryFocusedItemChanged

This procedure enables you to track the whereabouts (by Cell) of the cursor in the current record.

The format of this Procedure is:

Format: DataEntryFocusedItemChanged(PrevFocusedIndex: Integer; FocusedIndex: Integer);

The elements that make up this Procedure are:

PreviousFocusedIndex: The index number of the previous focused field

FocusedIndex: The index number of the current focused field

Let's see this in action. Create a new script and copy the following

```

procedure DataEntryFocusedItemChanged(PrevFocusedIndex: Integer; FocusedIndex:
Integer);
begin
  DataEntrySetLabel('Enter value for column ' + inttostr(FocusedIndex + 1));
end;

begin
  {create the data entry form}
  DataEntryCreate('Script Data Entry',500,720);
  {Create columns editor types shown below}
  DataEntryColumnCreate('Lookup',80,'LOOKUP',1004); {0}
  DataEntryColumnCreate('Text Editor',150,'TEXT'); {1}
  DataEntryColumnCreate('Date Editor ',80,'DATE'); {2}
  DataEntryColumnCreate('Calc Editor',60,'CALC'); {3}

```

```
DataEntryColumnCreate('Time Editor',80,'TIME'); {4}  
DataEntryColumnCreate('Curr Editor',80,'CURRENCY'); {5}  
DataEntryColumnCreate('Check Editor',80,'CHECKBOX');{6}  
DataEntryColumnCreate('Spin Editor',80,'SPIN'); {7}  
DataEntryShow;
```

end.

If you create a new record then the Screen Label will tell you to add data into the focused Column.

18 17. User-Defined Tables

Ostendo provides you with the facility to you to create your own Database Tables in Ostendo. These created Tables will then allow you to:

- Maintain your data using standard Add, Change, Delete, and Inquiry functions
- Have data maintenance screen appear in the standard Ostendo Menu.
- Fully integrate the Table(s) within Ostendo's standard database structure
- Be able to produce Reports, Views, Charts, and Inquiries
- Allow access to Screens, Reports, etc using standard User access routine

In this Training exercise we will create a 'Loan Register' in which you enter the Loan Equipment directly or copy information from the Company Asset as set up via General>Company Assets. Against these entries we will monitor who has the equipment; and for how long.

18.1 Preparation

There will be two Tables to cover the Loan Register.

1.1. The first table defines the Loan Equipment itself

Equipment Master

Equipment Name - Selected from Ostendo's Company Asset
 Equipment Type - Copied from the selected Company Asset
 Status - Copied from the selected Company Asset
 Description - Copied from the selected Company Asset
 Serial Number - As entered or from Company Asset
 Notes

1.2. Linked to the above is the ongoing history.

Loan History

Loan Type - (Planned, Loaned, Returned)
 Loaned To Type - (Employee, Customer, Supplier)
 Name: Name of Employee, Customer or Supplier
 Notes:
 Date Out - Planned
 Date In - Planned
 Date Out - Actual
 Date In - Actual

18.2 Creating the Tables in Ostendo

Go into [File>System Configuration>User Defined Tables](#) and ensure that the 'New' Radio Button is selected.

2.1. Equipment Master.

In the 'Name' field enter **EQUIPMENTMASTER** then click the 'Create Table' button. You will see that a Table called **OSTDEF_EQUIPMENTMASTER** Click the 'Add' and add the following fields to this Table. At this point we are not validating any entries but simply defining the field Names, Types and Sizes

Name	Type	Size
EQUIPNAME	Varchar	30
EQUIPTYPE	Varchar	20

EQUIPSTATUS	Varchar	20
EQUIPDESCRIPTION	Varchar	50
SERIALNUMBER	Varchar	30
NOTES	BLOB	0

2.2. Loan History

Now click on the 'New' radio button and create the **LOANHISTORY** table with the following details

Name	Type	Size
LOANTYPE	Varchar	20
LOANEDTOTYPE	Varchar	10
LOANEDTONAME	Varchar	50
LOANNOTES	BLOB	0
DATEOUTPLANNED	Date	0
DATEINPLANNED	Date	0
DATEOUTACTUAL	Date	0
DATEINACTUAL	Date	0

Close 'User Defined Tables' and click on **General>Reports>Full Listing of Tables** and print out the above two tables. You will find that other (control) fields have been added to the created Tables.

18.3 Generating the Edit View

We will now link the above two tables together in addition to generating the Edit Views.
Preparation and Generation

Go back into **File>System Configuration>User Defined Tables** and 'check' the 'Existing' radio button then select **OSTDEF_EQUIPMENTMASTER** from the drop-down list.

In the displayed panel click on the 'Generate Edit View' tab and enter the following

Name: This is the name of the Edit View. Therefore enter (say) '**Loan Equipment**'

Title: This is the name that will appear in the Title Bar of the generated View. Enter (say) '**Equipment**'

Script: We'll come back to this later.

Lines Table: Select **LOANHISTORY** from the drop-down list

List and Detail Caption: Against each field enter a caption that will appear against the field in the Edit screen. For example

Name	Caption
EQUIPNAME	Equipment Name
EQUIPTYPE	Type
EQUIPSTATUS	Status
EQUIPDESCRIPTION	Description
SERIALNUMBER	Serial Number
NOTES	Notes

Click on **Line Captions** tab and enter a caption relating to the Loan History. For example

Name	Caption
LOANTYPE	Loan Type
LOANEDTOTYPE	Loaned To Type
LOANEDTONAME	Loaned To Name
LOANNOTES	Notes
DATEOUTPLANNED	Planned Date Out
DATEINPLANNED	Planned Date In
DATEOUTACTUAL	Actual Date Out

DATEINACTUAL Actual Date In

Now click the **'Generate'** button to generate the Edit View. You will be asked if you wish to run the Edit View immediately. Click the **'Yes'** button to see the view with the 3 tabs (**List**, **Detail**, and **Lines**). Click the **'Add'** button and create a new Loan Equipment record. You will notice that there is no data validation, etc as we have not yet created a script to support this Edit Screen.

You may also click on the **'Lines'** tab and add Loan History records.

Report and View developer

If you now go to **File>Reporting Configuration>Report and View Developer** you will see that an Edit View has been created for the **Loan Equipment** comprising of a Master Query for the **EQUIPMENTMASTER** and a sub query for the **LOANHISTORY**.

Menu Item

If you now go to **Custom>Edit Views** you will see that the Edit View is in the list of Edit Views. User Security access can be applied to this View if required.

18.4 The Script

The previous sections allow you to create and maintain your data without any checks or dependencies. Of course you would need to incorporate these to avoid problems such as having Duplicates, validating data with current Ostendo data, preventing deletion if other records are linked to these tables, etc.

In these exercises we will:

- Prefill fields from the Company Asset when creating a new record
- Create a drop-down lookup against **EQUIPNAME** linked to the Company Asset table. The selection will automatically pull down other data from the selected record
- Create a drop-down in **LOANEDTONAME** relating to the selection made under **LOANEDTOTYPE**
- Check Mandatory Fields to see if they contain data
- Validate a field content

1. Creating and linking the Script

Let's start off by first creating the base script and then progressively add the conditions.

Go into **File>Custom Scripts** and create a new script called (say) **'LoanEquip'**. In the **'Detail'** tab select a **Style** of **'Edit View'** from the drop-down list and **'Save'** the record.

If you now click on the **'Script'** tab you will see that it is pre-populated with procedures that could be used. We will come back to these in the following exercises. For now exit the Custom Scripts and go to **File>Reporting Configuration>Report and View Developer** and select the Edit View you created above. Click on the **'Master Settings'** tab and – in the drop-down of field **Script Name** – select your script **'LoanEquip'**.

Save the record and return to your **'LoanEquip'** script under **File>Custom Scripts** and go to the **Script** tab so that we can now start to define the detailed activities.

2. Selecting a Company Asset

Let's start by linking field **EQUIPNAME** in the **EQUIPMENTMASTER** table to Ostendo's Company Assets

Go down to the end of the script to the final Begin and End lines and amend this to read:

```
Begin
  DetailLookup('Equipment_Name',1053);
End.
```

Where 'Equipment_Name' is the display name of database field EQUIPNAME and 1053 is the Scripting Lookup Number for Company Assets

If you now return to the Edit View and 'Add' a new record you will see that field Equipment Name now has a drop-down from which you can select the Company Asset

3. Copy selected Asset fields

We will now populate the remaining fields based upon the Company Asset selected.

Go down to procedure **DetailValidate**. This procedure allows you to carry out actions against a 'Detail' Level field based upon the entry in the nominated field. Enter the following against this procedure. This will populate the Status, Description, and Type fields from the Asset Master based upon the selection made in the 'Name' field

```
procedure DetailValidate(DisplayValue: Variant; AField: TField);
begin
  if (AField.fieldname = 'Equipment_Name') then
    begin
      Detailquery.fn('Status').AsString := GetSQLResult('Select ResourceStatus from ResourceMaster
where (ResourceName = '' + DisplayValue + '') and (ResourceType = "Asset")');
      Detailquery.fn('Description').AsString := GetSQLResult('Select AssetDescription from
ResourceMaster where (ResourceName = '' + DisplayValue + '') and (ResourceType = "Asset")');
      Detailquery.fn('Type').AsString := GetSQLResult('Select AssetType from ResourceMaster where
(ResourceName = '' + DisplayValue + '') and (ResourceType = "Asset")');
    end;
  end;
```

Points to note:

- The above Dataset field name is NOT the field name in the database (**EQUIPDESCRIPTION**) but the re-defined name in the Edit View (**Description**)
- The entry is also case-sensitive therefore enter '**Description**' and not **DESCRIPTION**

Go to **Custom>Edit Views** and run the View. If you click the '**Add**' button and select a Company Asset in the 'Equipment Name' field and exit the field you will see that the Type, Status, and Description fields are prefilled with data from the Company Asset.

Complete the remaining fields as required and '**Save**' the Record

4. Create a drop-down lookup against LOANTYPE and LOANEDTOTYPE

We will now define the options available in a drop-down against Loan Type and Loaned To Type fields in the Loan History entry screen.

Go down to the end of the script to the final Begin and End lines and add the following

```
LinesValuesCombo('Loan_Type','Planned,Loaned,Returned',True);
LinesValuesCombo('Loaned_To_Type','Employee,Customer,Supplier',True);
```

This section of your script should now look like this

Begin

```

DetailLookup('Equipment_Name',1053);
LinesValuesCombo('Loan_Type','Planned,Loaned,Returned',True);
LinesValuesCombo('Loaned_To_Type','Employee,Customer,Supplier',True);

```

End.

If you now return to the Edit View you will see that fields **Loan Type** and **Loaned To Type** have drop-downs from which the above selections are available

5. Create a related drop-down

This procedure defines what drop-down will be presented dependent on a selection made in another field. In this example we will define that the Loan History field **LOANEDTOTYPE** contains values of **Employee**, **Customer**, **Supplier** and that the drop-down against field **LOANEDTONAME** relates to the selection made

Go down to procedure **LinesFocusedItemChanged**. And enter the following

```

begin
  if (FocusedFieldName = 'Loaned_To_Name') then
    begin
      if linesquery.fn('Loaned_To_Type').AsString = 'Employee' then
        Begin
          LinesLookup('Loaned_To_Name',1040);
        end;
      if linesquery.fn('Loaned_To_Type').AsString = 'Customer' then
        Begin
          LinesLookup('Loaned_To_Name',1015);
        end;
      if linesquery.fn('Loaned_To_Type').AsString = 'Supplier' then
        Begin
          LinesLookup('Loaned_To_Name',1001);
        end;
      end;
    end;
  end;
end;

```

This states that if the cursor is placed in field **'Loaned_To'** then look at the current entry in field **'Loaned_To_Type'** and show the relevant drop-down

6. Validate the content of a Field

In this exercise we will look at validating the actual content being entered. This uses the procedure **LinesValidate** if in the Lines tab or **DetailValidate** if in the Detail Tab. For the purpose of this exercise we will validate that the **DATEOUTPLANNED** date (re-defined as **Planned_Date_Out**) in the Line record is not earlier than the System Date. This will simply return a message but allow you to continue.

Enter the following under Procedure **LinesValidate**:

```

procedure LinesValidate(DisplayValue: Variant; AField: TField);
begin
  if AField.fieldname = 'Planned_Date_Out' then
    begin
      if DisplayValue < date then
        begin
          Showmessage('Date is before today');
        end;
    end;
end;

```



```
end;  
end;  
end;
```

Try running the Edit View and entering a date prior to the system date.

7. Check Mandatory Fields to see if they contain data

The easiest way to do this is to use **File>System Configuration>Required Fields**. Therefore add a record to cover the above **Planned_Date_Out** field.

If you now run the Edit View without entering a Date then you cannot save the record

18.5 Reports and Views

Having created and maintained data in the above Tables you will want to print Reports and Views from within the Edit View grid. Here are a couple of simple examples:

1. Loan History Report

We will create a simple Report that lists Loan History against selected Loan Equipment.

Go into **File>Reporting Configuration>Report and View Developer** and click the **'Add'** button and create a new report called (say) **Loan History**.

In the Master Settings panel enter the following information

Include in Main Menu: Leave 'Unchecked'
Name: Leave as '**Loan History**'
Menu Order: Leave as 0
Category: select '**General**'
Type: Leave as '**Report**'
Specific Screen: Leave blank
Report File Name: Leave as '**Loan History**'
Archive: leave blank

Note: The Category is a mandatory field. However, because the 'Include in Main Menu' is unchecked the Report will not show under the General>Reports menu

Add this Query into the Master Data area

```
Select * from OSTDEF_EQUIPMENTMASTER
```

Now click on the Detail Queries tab and enter the following query against Query#1

```
Select * from OSTDEF_LOANHISTORY where HEADERSYSUNIQUEID =  
:SYSUNIQUEID
```

Return to the **'Master Settings'** tab and click on the **'Edit'** Button and create a report layout to suit your specific requirements. Here is an example of the finished layout:

ReportTitle: ReportTitle1							
Loan History Report							
MasterData: MasterData1							
Equipment Name	Type	Status	Description	Serial No			
[MD_"EQUIPNAME"]	[MD_"EQUIPTYP"]	[MD_"EQUIPSTAT"]	[MD_"EQUIPDESCRIPTION"]	[MD_"SERIALNUMBER"]			
Loan Type	Loaned To	Name	Planned - Out	Planned - In	Actual - Out	Actual - In	
[DD_1_"LOANTYPE"]	[DD_1_"LOANEDTOTY"]	[DD_1_"LOANEDTONAME"]	[DD_1_"DAT"]	[DD_1_"DAT"]	[DD_1_"DAT"]	[DD_1_"DAT"]	
DetailData: DetailData1							
PageFooter: PageFooter1							

Which would produce

Loan History Report							
Equipment Name	Type	Status	Description	Serial No			
Fork Lift	Plant	Active	3 <tr Fork Lift Truck	FL3225			
Loan Type	Loaned To	Name	Planned - Out	Planned - In	Actual - Out	Actual - In	
Returned	Supplier	Electrical Power Company	01/04/2010	02/04/2010	01/04/2010	03/04/2010	
Planned	Customer	Green Fingers Maloy Ltd	15/04/2010	30/04/2010	30/12/1899	30/12/1899	
Equipment Name	Type	Status	Description	Serial No			
Computer	Computer	Active	Acer PC				
Loan Type	Loaned To	Name	Planned - Out	Planned - In	Actual - Out	Actual - In	
Returned	Employee	Jane Steel	02/04/2010	09/04/2010	02/04/2010	09/04/2010	
Loaned	Employee	John Redmond	17/04/2010	19/04/2010	30/12/1899	30/12/1899	

Selection Conditions

You may wish to add conditions to the Report to give the option to select a range of equipment. To do this you would create a couple of conditions linked to Condition Type 5000

The format of this type of condition is

DisplayName;ParamDef;TableName;Fields;Captions;ReturnedField;OrderBy where:

- DisplayName** is the text that you wish to appear on the parameter entry screen
- ParamDef** has three parts:
 - KeyField** is the field name in the Master Query against which you making the comparison
 - Condition** is the condition that you are applying to the parameter
 - TheStoredValue** is a unique reference that you give this selected parameter
- TableName** is the name in the Table containing your selection
- Fields** are the Field Names from the selected Table that you wish to be displayed
- Captions** are the names that equate to the Fields and will be displayed in the extracted column headings
- ReturnedField** is the field in the selected Table from which data will be returned and

populate the above 'DisplayName' Variable

OrderBy is the sort order by which the selected records will be displayed

Condition Index For this style of Parameter entry this is always 5000

In our example we would create a couple of Parameters to select a Range of equipment as follows:

```
Equipment ID From;EQUIPNAME>=:EQUIPNAME;OSTDEF_EQUIPMENTMASTER
;EQUIPNAME;Equipment;EQUIPNAME;EQUIPNAME
```

```
Equipment ID To;EQUIPNAME<=:EQUIPNAME;OSTDEF_EQUIPMENTMASTER
;EQUIPNAME;Equipment;EQUIPNAME;EQUIPNAME
```

The Conditions Code will be **5000** in both instances

Preview the Report to ensure that it runs OK.

The next step is to get the report to appear under the '**Reports**' button on the above Edit screen. To achieve this we should amend the script created above as follows:

Go to your script and add the AddReportMenuItem line to give the following

```
Begin
AddReportMenuItem('Loan History');
DetailLookup('Equipment_Name',1053);
LinesValuesCombo('Loan_Type','Planned,Loaned,Returned',True);
LinesValuesCombo('Loaned_To_Type','Employee,Customer,Supplier',True);
End.
```

If you now go into the Edit View and click on the '**Reports**' Button you will see that this entry appears. What we now need to do is actually run the Report when the report is selected. As you can have many Reports under this Button we need to differentiate the individual Reports. Each report under the Report button is given a Menu Index and you would refer to this when running the Report within the script. Let us first show you the Report Menu Index. Go to Procedure **ReportMenuItemClicked** and enter the following:

```
procedure ReportMenuItemClicked(MenuIndex: Integer);
begin
Showmessage(inttostr(MenuIndex));
end;
```

If you run the Edit View and click on the Report Button then click on 'Loan History' you will see a message telling you the Menu Index number. As this is the only one in the list the Index Number will be zero. Using this number, amend the above procedure as follows:

```
procedure ReportMenuItemClicked(MenuIndex: Integer);
begin
If MenuIndex = 0 then
begin
OstendoReport('Loan History');
end;
end;
```

If you now click on '**Loan History**' under the **Reports** button then your report will be run

2. Loan Equipment Analysis

We will now create a simple Analysis View showing all the Loan Equipment

Go into **File>Reporting Configuration>Report and View Developer** and click the **'Add'** button and create a new Analysis called (say) **Loan Equipment**.

In the Master Settings panel enter the following information

Include in Main Menu: Leave 'Unchecked'

Name: Leave as **'Loan Equipment'**

Menu Order: Leave as 0

Category: **General**

Master Key Field: Leave blank

Detail Key Field: Leave blank

Title: Enter **'Loan Equipment'**

Merge Word Document: leave blank

Merge Data File: leave blank

Add this Query into the Master Data area

Select EQUIPNAME as "Equipment", EQUIPTYPE as "Type", EQUIPSTATUS as "Status", EQUIPDESCRIPTION as "Description", SERIALNUMBER as "Ser No", SYSUNIQUEID as GridUniqueIndex from OSTDEF_EQUIPMENTMASTER

Click the **'Preview'** button to ensure that the Analysis View runs OK.

The next step is to get the Analysis View to appear in the Edit View. You can place it under the **'Reports'** Button by repeating the above steps or you can place it under the **'Related'** button using the following steps:

Begin

AddRelatedMenuItem('Loan Equipment');

AddReportMenuItem('Loan History');

DetailLookup('Equipment_Name',1053);

LinesValuesCombo('Loan_Type','Planned,Loaned,Returned',True);

LinesValuesCombo('Loaned_To_Type','Employee,Customer,Supplier',True);

End.

If you now go into the Edit View and click on the **'Related'** Button you will see that this entry appears. What we now need to do is actually run the Analysis View when it is selected. As with the Reports button you can have many entries under this Button and it works in the same way.

Therefore, using Index number zero add the following under Procedure **RelatedMenuItemClicked**

```
procedure RelatedMenuItemClicked(MenuIndex: Integer);
```

```
begin
```

```
  If MenuIndex = 0 then
```

```
    begin
```

```
      OstendoAnalysis('Loan Equipment');
```

```
    end;
```

```
end;
```

If you now click on **'Loan Equipment'** under the **Related** button then your Analysis View will be run.

18.6 User-Defined Menus

By default your User-defined Edit View will appear under 'Custom' on the top toolbar. You can locate this under any of the main Ostendo Toolbar headings. In this exercise we will place this under 'General'

Go into **File>System Configuration>User defined Menus** and create a new record containing the following

- **Custom Menu Name:** From the drop-down list select the Edit View you created above (Example: **Loan Equipment**)
- **Custom Menu Type:** This is display only and should be **Edit View**
- **Menu Category:** From the drop-down list select **General**
- **Sub Menu Name:** Type in a name (example **Loan**) under which **Loan Equipment** will be displayed

'Save' the record and then go to General on the top toolbar of Ostendo where you will see **Loan** in the list. If you select this then you will see **Loan Equipment** available under this sub menu

18.7 Show Help

You may wish to create your own Help associated with this Edit View. One way is to simply use the 'Related' Button as follows.

The next step is to get the Analysis View to appear in the Edit View.

Add the following line to the final Begin and End
 AddRelatedMenuItem('Help Document');

This should now look like this

```

Begin
  AddRelatedMenuItem('Loan Equipment');
  AddRelatedMenuItem('Help');
  AddReportMenuItem('Loan History');
  DetailLookup('Equipment_Name',1053);
  LinesValuesCombo('Loan_Type','Planned,Loaned,Returned',True);
  LinesValuesCombo('Loaned_To_Type','Employee,Customer,Supplier',True);
End.

```

Now you need to add this to the RelatedMenuItemClicked procedure using Index 1. The whole procedure would now be

```

procedure RelatedMenuItemClicked(MenuIndex: Integer);
begin
  if MenuIndex = 0 then
    begin
      OstendoAnalysis('Loan Equipment');
    end;
  if MenuIndex = 1 then
    begin
      Run('D:\Ostendo\Help\Help - Equipment Loan.doc');**
    end;
end;

```

** where 'D:\Ostendo\Help\Help - Equipment Loan.doc' is the full path to the Help document

If you now click on 'Help' under the **Related** button then your Help Document will be returned.

19 18. Accounting Interface

Ostendo allows you to manage the operational activity of your business and keep your existing accounting software. Business operations typically fall outside the scope of financial systems. Ostendo integrates these crucial activities into your business, ensuring that you have real time information for decision-making support. Ostendo therefore provides financial information covering:

- Customer Deposits
- Sales and Job Invoicing
- Customer Payment receipts
- Bank Deposits
- Ongoing Work In Progress and Job Costing
- Inventory Control
- Purchase Orders raised.
- Goods Received and Invoice Matching

Information from these areas is sent to an interface program created by Development-X from where it is posted to a 3rd Party Accounting system.

19.1 MYOB

Ostendo's Accounts Integration function provides facility to have

- Instant integration with MYOB using preset Account Codes
- Comprehensive mapping options for more detailed Accounting Structures

The following will be covered in this document:

- Overview
- Setup steps
- Process a single transaction from Ostendo to MYOB
- Base Mapping to pre-defined activities
- 'T' Charts of Journals generated from within Ostendo
- Additional Sales, Labour and Inventory Mapping

19.1.1 Overview

1. Coverage

Ostendo covers the following functional areas.

- Sales order process comprises Customers, Orders, Delivery, Invoicing, Payment receipts and Banking. Ostendo creates GL Journals and these are posted directly to MYOB's GL therefore MYOB AR is not used
- Purchase Order process comprising Suppliers, Purchase Orders, Goods Receipt, Invoice Receipt and matching. In this process Suppliers are maintained in Ostendo and any new additions or changes in Ostendo will update the Supplier records in MYOB. Valid Purchase Order Receipts and Invoices are posted to MYOB's AP for payment processing
- Assembly Orders are fully maintained in Ostendo and the appropriate Job Costs are recorded and posted to MYOB GL
- Inventory issues, receipts and valuations are fully maintained in Ostendo and the appropriate Journals are posted directly to MYOB GL

2. Data Flow



Ostendo has a series of Cost Centres against which financial activity is recorded. These are combined to form various Financial Journals covering all activities across Ostendo. A screen is available in Ostendo from which you send a batch of Journals to a separate '[Ostendo to MYOB Link](#)' program. It is within this routine that Ostendo's Cost Centres are converted to MYOB Account Codes.

The updated results (and possible errors) is returned to the 'Ostendo to MYOB Link' from where it can be returned to Ostendo.

19.1.2 Setup

1. Ostendo

Out-of-the-Box Ostendo contains a list of pre-defined areas and their associated Cost Centres. We will expand on these later in this exercise. For now, therefore, no further setup action is required.

The first step is to go to File>System Configuration>Systems Settings and click on the 'Accounting Link' tab. Ensure that the 'Accounting Link Style' is set to 'MYOB Link'

2. MYOB

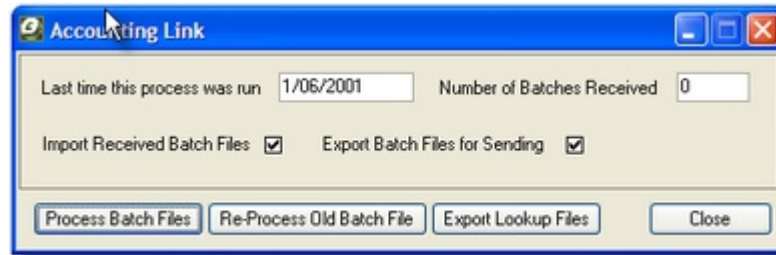
The only thing that is essential is that the correct ODBC Driver for the version of MYOB used is installed. The following versions have been tested and approved for the Ostendo to MYOB interface program

<u>Australia</u>	<u>Version</u>	<u>ODBC Driver Version</u>
MYOB Accounting	15, 16	6.0.14
MYOB Accounting	17	7.0.11
MYOB Accounting Plus	15, 16	6.0.14
MYOB Accounting Plus	17	7.0.11
MYOB Premier	9, 10	6.0.14
MYOB Premier	11	7.0.11
MYOB Premier Enterprise	4	6.0.14
MYOB Premier Enterprise	5	7.0.11
<u>New Zealand</u>	<u>Version</u>	<u>ODBC Driver Version</u>
MYOB Accounting	15, 16	6.0.14
MYOB Accounting	17	7.0.11
MYOB Accounting Plus	15, 16	6.0.14
MYOB Accounting Plus	17	7.0.11
MYOB Premier	9, 10	6.0.14
MYOB Premier	11	7.0.11
MYOB Premier Enterprise	4	6.0.14
MYOB Premier Enterprise	5	7.0.11

3. Ostendo to MYOB Link

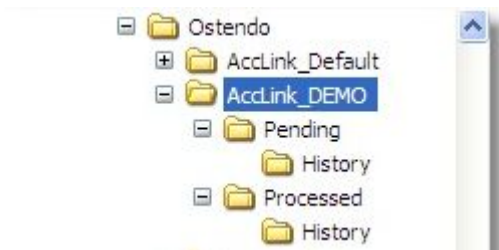
3.1. Export Lookup Files

Before we address the setup in the '**Ostendo to MYOB Link**' routine you should first go into Ostendo and go to screen **File>Accounting Link**



On that screen click on the '**Export Lookup Files**' button. This will carry out the following actions:-

1. Under the Ostendo folder it will create folder **ACCLINC_DEMO** where DEMO is the name of the company that you are currently logged into. Under that folder you will see two sub-folders **Pending** and **Processed**. Each of these has a sub-folder called **History**



2. Directly under the '**Pending**' folder you will see two files

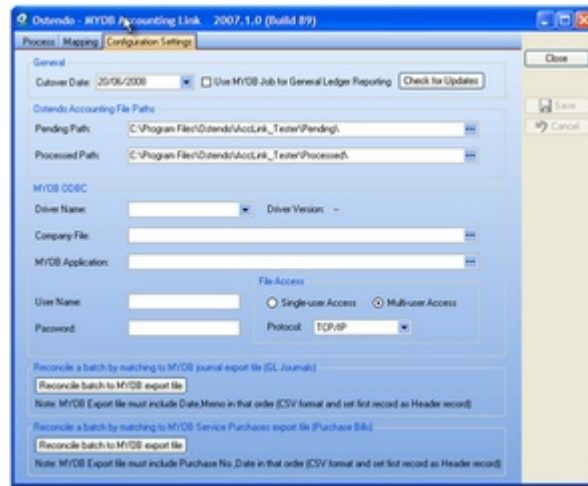
- Costcentre.lup
- Taxcodes.lup

These 'Lookup Tables' will be referenced during this setup

3.2. Set Up the Accounting Link

This section describes the fields in the Accounts Interface routine and the activities required Start up the Ostendo to Accounting Link program and you will asked to set up the setting to point to both Ostendo and MYOB.

Panel 3 - Configuration Settings panel



This panel requires you to point this ‘**Ostendo to MYOB Link**’ routine to both Ostendo and MYOB. The following fields are shown in this screen

- **Cutover Date:** This is used to allow you to define the date when Ostendo Transactions will begin posting to MYOB. Transactions prior to this date will be received by this function but will not be passed onto MYOB.

Ostendo Accounting File Paths

- **Receive Path:** This defines where the Batch details received from Ostendo will be stored. Click on the 3 dots icon and select the location of the folder. This will be the **Pending** folder you set up above.
Example **c:\Program Files\Ostendo\Acclinc_DEMO\Pending**
- **Send Path:** This defines where the details of the Batch - after updating MYOB - will be stored. Click on the 3 dots icon and select the location of the folder. This will be the **Processed** folder you set up above.
Example **c:\Program Files\Ostendo\Acclinc_DEMO\Processed**
- **Use MYOB Job for General Ledger Reporting:** If you use Jobs in MYOB for General Ledger reporting then click on this checkbox. This will then display the MYOB Jobs in panel 2 for mapping purposes.

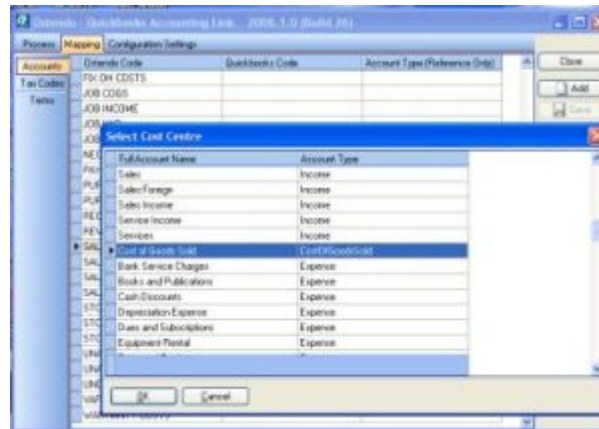
MYOB Integration

- **Company File:** This defines the location of the MYOB Company database. Click on the 3 dots icon and select the location of the database
- **MYOB Application:** This defines where the MYOB application program (Example: C:\Premier10\Myobp.exe) is located. Click on the 3 dots icon and select the location of the program
- **Username:** Enter the MYOB username (Example: Administrator) required to access the Company File
- **Password:** Enter the password used with the above MYOB username that is required to access the Company File
- **Single User Access:** This must be set up to reflect the current way that MYOB is set up in your company
- **Multi-User Access:** This must be set up to reflect the current way that MYOB is set up in your company
- **Protocol:** This must be set up to reflect the current way that MYOB is set up in your

company

Panel 2 - Mapping

This screen requires that you map Ostendo's Cost Centres and Tax Codes to MYOB's Account Codes and Tax Codes.



Step 1. The Ostendo Cost Centres will be displayed in the left-hand column. In the right-hand column select the equivalent MYOB Account codes from the drop-down list.

Step 2. Click on the 'Tax Codes' tab down the left side of the screen to display the Tax Codes screen. Click the 'Add' button and select a Tax Code. In the right-hand column select the equivalent MYOB Tax Code from the drop-down list. Repeat for each Ostendo Tax Code

You have now completed the setup.

Note: Any time you add Cost Centres or Tax Codes you should go to **File>Accounting Link** in Ostendo and download the updated Cost centres and/or Tax Codes then match the new Codes in the Ostendo to Accounting Link program.

19.1.3 Create and post your first Journal

We will create a simple stock adjustment in Ostendo and process that transaction through to MYOB. If you go into **Pricing>Item Costing** and select Item **100-2002** you will see that it has a current Standard Cost of **\$0.06**. We will receive **100** of these into stock

1. Create a Stock Adjustment

Go into **Inventory>Inventory Adjustments** and click the 'Add' button and 'Save' the record. Click on the 'Detail' tab and then the 'Add' button to create a new Line. Enter the following information:

Item Code: **100-2002** (selecting this will prefill the other fields)
Adjustment Type: from the drop-down select '**RECEIPT**'
Adjustment Qty (+/-): Enter **100**

Go back to the 'Detail' tab and click on the '**Post all Adjustments**' button. This will carry out the adjustment and create the journal. To view the journal go into **General>Reports** and select '**Financial Batch Reports**'. Enter the following parameters:

Transaction Status: Select **'Ready to Send'**
From Transaction Date: Select Today's Date
To Transaction Date: Select Today's Date
From Batch No: Leave Blank
To Batch No: Leave Blank
Exclude Conditions: 'Check' this checkbox

Click the **OK** button bring the report back to the screen

FINANCIAL BATCH REPORT						
Grouped by Status & Type						Development-X Limited
Transaction Status: Ready to Send						
Transaction Type: Inventory						
Trans Date	Batch No	Debit Cost Centre	Credit Cost Centre	Tax Amt	Tax Code	Amount
24/09/2008		STOCK	STOCK ADJUST	\$6.00		\$6.00
Source: (ADJUSTMENT) Type: RECEIPT Inventory Adjustment: 14						
Totals for: Inventory				\$6.00		\$6.00
Totals for Status: Ready to Send				\$6.00		\$6.00

You will see that the generated journal has a status of **'Ready to Send'** (i.e. ready to send to the **Ostendo to MYOB Link** program. The journal itself contains the following info that we will focus on

Debits Cost Centre: **STOCK**
 Credits Cost Centre: **STOCKADJUST**

2. Generate a Posting Batch

The next step is to send the created Journal to the **'Ostendo to MYOB Link'** program.

To do this go into **File>Accounting Link** and 'Uncheck' the **Import Received Batch Files** checkbox and click the **'Process Batch Files'** button. This will gather all Journals with a **'Ready to Send'** status and create a single Batch file. This file will be stored under the **'Pending'** folder that you created above.

3. Updating MYOB

Go into the **Ostendo to MYOB Link** program and select the **'Mapping'** tab. If you haven't already done so then ensure that a MYOB GL Account is associated with Cost Centres **STOCK** and **STOCKADJUST**.

Click on the **'Process'** tab where you should see that **'1 file is waiting to be processed'**. Click on **'Process'** button and the Link program will update MYOB. You will see that progress in the main panel. If any errors are found then they will be displayed here.

You will also find that the Batch file that was in the 'Pending' folder has now moved to **Pending>History**. Also the Processed Folder will now contain all the processed information about the Batch File

4. MYOB

If you now go into MYOB you can drill down on the Account Codes or look in the Audit trail where you will see the postings

5. Update Ostendo

The final step in this short exercise is to return the results of the Batch that was sent to the Ostendo to MYOB Link program

To do this go into **File>Accounting Link** and make sure that the **Import Received Batch Files** checkbox is 'checked' and click the **'Process Batch Files'** button. This time it will get the Batches currently held in the **'Processed'** folder of the Link program and bring them back into Ostendo. This action will also move the batches to **Processed>History**.

If you now go into **General>Reports** and select **'Financial Batch Reports'**. Enter the following parameters:

Transaction Status: Select **'Transaction Valid'**

From Transaction Date: Select Today's Date

To Transaction Date: Select Today's Date

From Batch No: Leave Blank

To Batch No: Leave Blank

Exclude Conditions: 'Check' this checkbox

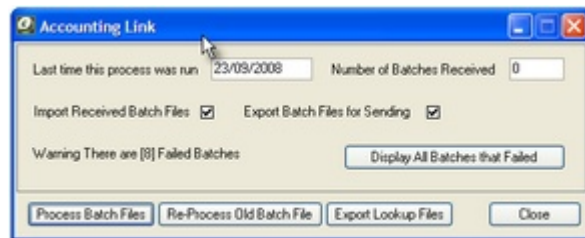
Providing the Transaction posted Ok you should see the updated Journal. If the Batch had errors then it should be processed as described below

That concludes the process flow for a single transaction. However, let us take a closer look at the actions required if any error where found

6. Batches with Errors

6.1. Failed Batch

If there are Batches with errors then the Accounts Link screen will now display an extra button **'Display all Batches that Failed'**.



If you click on this button then all the Batches that have errors will be displayed along with drill-down to show the specific Transactions that contain errors. Action should now be taken to correct the problem(s) that caused the error

Having corrected the errors click on the **'Display All Batches that Failed'** button. Select the Batch Number and then click the **'Re Post Batch'** button. This action will reset those transactions in the Batch whose status is **'Transaction Invalid'** to **'Ready to Send'**

These will be picked up and included in a new batch when you click on the **'Process Batch Files'** button

6.2. Lost Batch

Instances may arise where - for whatever reason - the batch sent to the Ostendo to MYOB Link had not been returned with a Batch Status update. Facility is provided where you can resend this

Batch if necessary. If you click on the '[Re-Process Old Batch File](#)' button then a panel will be presented for you to select and re-send the previously sent Batch.

19.1.4 Cost Centres

Ostendo uses the concept of Cost Centres to record costs from all areas of the product. These Cost Centres are user-defined and are directly linked to General Ledger Account Codes in MYOB via the [Ostendo to MYOB Link](#) program

1. Base Cost Centres

Ostendo comes with 'Base Level' settings in which pre-defined Cost Centres are ready to be mapped to MYOB GL Account Codes

Cost Centre Groups

Cost Centres can be grouped into logical areas. These relate to the Cost Areas such as

- Asset
- Liability
- Income
- Expense
- Cost of Sale
- Bank
- etc

Go into [General>Settings>Cost Groups](#) and enter a couple of Groups

Cost Centre Maintenance

If you go into [General>Cost Centres](#) you will see the base set of Cost Centres supplied with Ostendo. In the absence of any changes these Cost Centres will be used when integrating with MYOB.

2. Base Cost Centre Mapping

The above Base Cost Centres are linked to areas where costs are stored. Out-of-the-Box Ostendo contains a list of pre-defined areas and their associated Cost Centres. You can see these by going into [File>Financial Configuration>Cost Centre Mapping](#).

The pre-defined areas are:

Stock: All Stock across all Warehouses and Locations in your Company's Inventory is kept against this Cost Centre. If a Warehouse is allocated a specific Cost Centre then that will be used in preference to this default.

Stock Adjustment Variance: Stock Adjustments are 'received from' or 'issued to' a Cost Centre. This Cost Centre provides a global place for 'posting' the Costs from this function

Negative Adjustment Variance: Stock adjustment variances can occur whenever you receive Stock but the current stock levels are negative. In this case the Average Cost for the Item is not amended but the difference between the Average Cost and the Cost of this receipt is 'posted' to this Cost Centre

Revaluation of Stock Variance: Whenever the Stock is re-valued (Standard Cost or Average Cost) the difference between the old value and the new value is 'posted' to this Cost Centre

Stock Count Variance: Stock variances (plus or minus) as a result of a Stock Count are 'posted' to this Cost Centre.

Stock In Transit: Whenever Stock is moved from one Warehouse to another via the Inventory Transfer routine this Cost Centre will be used for that transfer.

Receipt Cost Var: Used if the Inventory Costing method is 'Standard Costing'. In that instance this stores the cost difference between the Assembly Order actual Cost and the Standard Cost of the Item.

Descriptor Expenses: This is the default Cost Centre used by Descriptors. It can be amended at Descriptor level to another Cost Centre and made specific to that Descriptor. This Cost Centre is then used as the 'allocated' Cost Centre during Purchase Receipts.

Catalogue Expenses: This is the default Cost Centre used by Supplier Catalogue Items. This Cost Centre is used as the 'allocated' Cost Centre during Purchase Receipts. If a Supplier Catalogue is allocated a specific Cost Centre then that will be used in preference to this default.

Warranty Costs: Where a Job Issue has it's 'Actual Issue' allocated to a Charge Style of 'Warranty' then the costs are 'posted' to this Cost Centre

Contract Costs: Where a Job Issue has it's 'Actual Issue' allocated to a Charge Style of 'Contract' then the costs are 'posted' to this Cost Centre

Direct Labour Costs: When an Employee Timesheet Transaction is made then the costs are 'posted' to this Cost Centre

Fixed Labour Overhead Costs: When an Employee Timesheet Transaction is made then the associated FOH costs - held against the linked Labour Code - are 'posted' to this Cost Centre

Variable Labour Overhead Costs: When an Employee Timesheet Transaction is made then the associated VOH costs - held against the linked Labour Code - are 'posted' to this Cost Centre

Purchases Received Not Invoiced: For any Purchase Orders that have been received via the 'Purchase Order Receipts' function are held in this Cost Centre until they have been 'matched' in the 'Purchase Order Invoicing' screen

Purchase Price Variance: Purchase Orders received via the 'Purchase Order Receipts' function contain a Purchase Price and this is 'posted' at that time. When it is matched against the 'Purchase Order Invoicing' there may be a Price difference. This difference is posted to this Cost Centre.

Creditors: Whenever a Purchase Invoice is received then the amount of that Invoice is 'posted' to this Cost Centre.

Assembly Work In Progress: For any Issues (Material, Labour, Descriptors, etc) that have been issued to Work In Progress for an Assembly Order the cost of that issue is 'posted' to this Cost Centre. Note: The cost of the Assembly Order is taken out of this Cost Centre whenever the Assembly Order is received into Inventory

Assembly Order Variance: If the Assembly Order has been completed and its status is 'Closed' then any subsequent Issues or Bookings to this Order will go direct to this Cost Centre rather than the Assembly Work In Progress Cost Centre.

Job Work In Progress: For any Issues (Material, Labour, Descriptors, etc) that have been issued to Work In Progress for a Job Order the cost of that issue is 'posted' to this Cost Centre. Note:

The cost of the Job Order is taken out of this Cost Centre whenever the Job Order is Invoiced.

Job Cost Of Goods Sold: Any Invoice raised against the Job Order will take the cost of the Invoice out of the 'Job Work In Progress' Cost Centre and add it to this Cost Centre.

Job Cost Variance: If the Job Order is 'Closed' then any subsequent Issues or Bookings to this Order will go direct to this Cost Centre rather than the Job Work In Progress Cost Centre.

Job Income: Any Income received that has been applied to a Job Invoice Line will be 'posted' to this Cost Centre.

Sales Lines Picked: Whenever a Sales Order Line is 'Picked' the cost of that line is moved from the source Cost Centre (Example: 'Stock' Cost Centre for Inventory Items) and moved to this Cost Centre.

Sales Cost Of Goods Sold: Whenever a Sales Order Line is 'Invoiced' the cost of that line is moved from the 'Sales Line Picked' Cost Centre to this Cost Centre.

Sales Income: Any Income received that has been applied to a Sales Invoice Line will be 'posted' to this Cost Centre.

Freight: Any Income received that covers Freight and has been applied to an Invoice will be 'posted' to this Cost Centre.

Debtors: Whenever an Invoice (or credit) is raised then the amount of that Invoice is 'posted' to this Cost Centre. Whenever a Deposit or Payments is matched to an Invoice then it is moved from this Cost Centre to the above Sales Invoice, Sales Freight, and Sales tax Cost Centres.

Contract Income: Any Income received that has been applied to a Contract Invoice Line will be 'posted' to this Cost Centre. Until it is 'posted' the income will reside in the 'Unapplied Deposits' or 'Unapplied Payments' Cost Centres

Deferred Cost Of Goods: The value of any Retention amount raised against a Job Order Invoice will take the cost of the Retention out of the 'Job Work In Progress' Cost Centre and add it to this Cost Centre.

Deferred Income: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Income deferred for future P & L recognition.

Deferred Freight: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Freight deferred for future P & L recognition

Deferred Tax: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Tax deferred for future P & L recognition

Un-Applied Deposits: Deposits received from a Customer will be 'posted' to this Cost Centre. When they are applied to an Invoice they will be moved to the 'Sales Income' Cost Centre

Un-Applied Payments: Payments received from a Customer will be 'posted' to this Cost Centre. When they are applied to an Invoice they will be moved to the 'Sales Income' Cost Centre

Bank: Upon validation that the Payment was banked (Deposit Slip status updated to 'Banked' then the amount is transferred from the 'Un-deposited Funds' Cost Centre and 'posted' to this Cost Centre

Payment Rounding: If the 'matched' payment amount is within a 'Rounding Tolerance' entered in that screen then the Payment record status is amended to 'Fully Applied'. The actual value of the Rounded amount is 'posted' to this Cost Centre,

Discounts Given: If a Discount Amount is entered during Payment Matching (Example: 'Prompt Payment' discount) then the value of this discount is 'posted' to this Cost Centre.

Undeposited Funds: Whenever a Bank Deposit Slip is raised (Status is 'Pending') the included payments are 'posted' to this Cost Centre.

POS End of Day Variance: Whenever a 'Z' Report is printed from the End Of Day processing in the Point of Sales function than if the Till Balance is at variance to the expected balance then the difference is posted to this Cost Centre

POS Shop Expenses: Whenever a Cash withdrawal or deposit is made within the POS function then the amount Withdrawn/Deposited is held in this Cost Centre

19.1.5 Base Mapped Cost Centres

The base mapped Cost Centres in any installation of Ostendo are:

Cost Accumulation Area	Linked Cost Centre
Stock	STOCK
Stock Adjustment Variance	STOCK ADJUST
Negative Stock Variance	NEG STOCK
Revaluation Stock Variance	REVALUE STOCK
Stock Count Variance	COUNT STOCK
Stock In Transit	STOCK INTRANSIT
Receipt Cost Variance	RECEIPT COST VAR
Descriptor Expenses	DESCRIPTOR EXPENSES
Catalogue Expenses	CATALOGUE EXPENSES
Warranty Costs	WARRANTY COSTS
Contract Costs	CONTRACT COSTS
Direct Labour Costs	DIRECT LABOUR COSTS
Fixed Labour Overhead Costs	FIX OH COSTS
Variable Labour Overhead Costs	VAR OH COSTS
Purchases Received Not Invoiced	PURCHASES RECEIPTS
Purchase Price Variance	PURCHASE PRICE VAR

Creditors	CREDITORS
Assembly Work In Progress	ASSEMBLY WIP
Assembly Order Variance	ASSEMBLY VAR
Job Work In Progress	JOB WIP
Job Cost Of Goods Sold	JOB COGS
Job Cost Variance	JOB VAR
Job Income	JOB INCOME
Sales Lines Picked	SALES PICKED
Sales Cost Of Goods Sold	SALES COGS
Sales Income	SALES INCOME
Sales Freight	SALES FREIGHT
Debtors	DEBTORS
Contract Income	CONTRACT INCOME
Deferred Cost Of Goods	DEFERRED COGS
Deferred Income	DEFFERED INCOME
Deferred Freight	DEFFERED FREIGHT
Deferred Tax	DEFFERED TAX
Un-Applied Deposits	UNAPPLIED DEPOSITS
Un-Applied Payments	UNAPPLIED PAYMENTS
Bank	BANK
Payment Rounding	PAYMENT ROUNDING
Discounts Given	DISCOUNTS GIVEN
Un-Deposited Funds	UNDEPOSITED FUNDS
POS End of Day Variance	POS EOD
POS Shop Expenses	POS EXPENSES

19.1.6 'T' Charts

If you go into [General>Reports](#) you will see the following one-page documents showing the flow through that process.

[Financial Flow - Jobs](#)
[Financial Flow - Sales](#)
[Financial Flow - Assembly](#)
[Financial Flow - Purchasing](#)

Each step within those flows use the following 'T' Charts. These are pre-defined in Ostendo and cover the various activities that take place.

Sales Issues

This covers issue of Items, Descriptors and Supplier Catalogue Items to a Sales Order

Debit	Credit
<i>SALES LINES PICKED</i>	*(For Inventory) <i>Cost Centre</i> **(For Descriptors) <i>Cost Centre</i> ***(For Catalogue Items) <i>Cost Centre</i>

* If the Warehouse from where the Item was picked does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Sales Invoice

This covers the production of the Invoice. It should be noted that the Cr values include Tax because Tax is generated in MYOB relating to the Tax Code passed along with the Journal

Debit	Credit
<i>SALES COGS</i> <i>DEBTORS</i>	<i>SALES LINES PICKED</i> <i>SALES INCOME</i> <i>SALES FREIGHT</i>

Purchase Order Receipt

This covers the production of the Invoice. It should be noted that the Cr values include Tax because Tax is generated in MYOB relating to the Tax Code passed along with the Journal

Debit	Credit
*(For Inventory) If using Standard Cost) Cost Centre RECEIPT COST VAR or If not using Standard Cost) Cost Centre **(For Descriptors) Cost Centre *** (For Catalogue Items) Cost Centre (For Direct Allocations) JOB WORK IN PROGRESS or ASSEMBLY WORK IN PROGRESS	PURCHASES RECEIVED NOT INVOICED

* If the Warehouse into which the Item is received does not contain a Cost Centre then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Purchase Price Variance

This covers situations where the Purchase Order has been received and - at some later time - the Invoice is matched against the Receipt and where the Invoice contains a different price to the receipt price. This difference is posted to a Purchase Price Variance Cost centre

Debit	Credit
(For Inventory) PURCHASE PRICE VARANCE *(For Descriptors) Cost Centre **(For Catalogue Items) Cost Centre (For Direct Allocations) JOB WORK IN PROGRESS or ASSEMBLY WORK IN PROGRESS	PURCHASES RECEIVED NOT INVOICED

* If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Purchase Order Invoice

The Purchase Invoice is received, matched against prior receipts, and approved.

Debit	Credit
PURCHASES RECEIVED NOT INVOICED	CREDITORS

Job Issues

Issues to Jobs comes from Stock Items, Descriptors, or Labour. These issues could be charged to the Job or can be designated as non-chargeable (covered by Warranty, a Contract, or simply making it non-chargeable when issuing it)

Debit	Credit
JOB WORK IN PROGRESS	*(For Inventory) Cost Centre
(Defined 'Non-Charge' at time of issue) Cost Centre	** (For Descriptors) Cost Centre
(Non-Charge Warranty) WARRANTY COSTS	*** (For Catalogue Items) Cost Centre
(Non-Charge Contract) CONTRACT COSTS	(Labour - broken down into) DIRECT LABOUR COST FIXED LABOUR OVERHEAD COST VARIABLE LABOUR OVERHEAD COST

* If the Warehouse from where the Item was picked does not contain a Cost Centre then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the Cost Centre matched to CATALOGUE EXPENSES will be used

Job Invoice

Upon creation of a Job Invoice 2 Journal transactions are created.

- Relating to the Costs of the Invoice
- Relating to the Expected Income

Debit	Credit
JOB COST OF GOODS SOLD	JOB WORK IN PROGRESS
Debit	Credit
DEBTORS	JOB INCOME * (For Tax) FREIGHT

* Tax is created within MYOB from the Tax Code passed through. I.e. The Job Income amount includes Tax

Job WIP Variance

If a Job's status is changed to 'Closed' then any residual costs for that Job remaining in Work In

Progress will be posted to a Job WIP Variance

Debit	Credit
(Invoice Job Style) <i>JOB COST OF GOODS SOLD</i> (If Job Style is 'No Invoice') <i>Cost Centre</i>	<i>JOB WORK IN PROGRESS</i>

Customer Payments

Payments received from a Customer can be deposited directly into the Bank or alternatively retained and 'batched' into a consolidated Bank Deposit Slip.

Debit	Credit
<i>BANK</i> or <i>UNDEPOSITED FUNDS</i>	<i>UNAPPLIED PAYMENTS</i>

Customer Deposits

Deposits received from a Customer can be deposited directly into the Bank or alternatively retained and 'batched' into a consolidated Bank Deposit Slip.

Debit	Credit
<i>BANK</i> or <i>UNDEPOSITED FUNDS</i>	<i>UNAPPLIED DEPOSITS</i>

Applying Payments

Payments are matched against Invoices raised against a Customer

Debit	Credit
<i>UNAPPLIED PAYMENTS</i>	<i>DEBTORS</i> <i>DISCOUNTS GIVEN</i> <i>PAYMENT ROUNDING</i>

Applying Deposits

Deposits are matched against Invoices raised against a Customer

Debit	Credit
<i>UNAPPLIED DEPOSITS</i>	<i>DEBTORS</i> <i>DISCOUNTS GIVEN</i> <i>PAYMENT ROUNDING</i>

Bank Deposits

Bank Deposits are applied Customer Payments and/or Customer Deposits that have been consolidated into a single Bank Deposit Slip

Debit	Credit
<i>BANK</i>	<i>UNDEPOSITED FUNDS</i>

Assembly Issues

Issues to Assembly Orders comes from Stock Items, Descriptors, or Labour.

Debit	Credit
<i>ASSEMBLY WORK IN PROGRESS</i>	*(For Inventory) <i>Cost Centre</i> **(For Descriptors) <i>Cost Centre</i> (Labour - broken down into) <i>DIRECT LABOUR COST</i> <i>FIXED LABOUR OVERHEAD COST</i> <i>VARIABLE LABOUR OVERHEAD COST</i>

* If the Warehouse from where the Item was picked does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

Assembly Receipts

Receipts from Assembly Orders always go into Inventory. The Journal created depends upon whether the Item being received uses Standard Costing or not

With Standard Costing

Debit	Credit

** (Warehouse) <i>Cost Centre</i> * <i>RECEIPT COST VAR</i>	<i>ASSEMBLY WORK IN PROGRESS</i>
--	----------------------------------

* If the Warehouse does not contain a Cost Centre against RECEIPT COST then the default Cost Centre matched to RECEIPT COST VAR will be used. The value of this variance comes from Cost of Receipt - Standard Cost.

No Standard Costing

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

** If the Warehouse does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

Assembly WIP Variance

Whenever an Assembly Order has its status changed to 'Closed' then any residual Costs for the Assembly Order are posted to a WIP Variance. Additional activity is carried out if the following conditions apply

- If the Assembly Order was for a Custom Product and that Product is still in stock then add the Cost to the Stock record
- If the Assembly Order was for a Custom Product and that Product has been withdrawn from stock but has not yet been Invoiced then add the cost to the Sales Order 'Pick but not invoiced' record
- If the Assembly Order was for a Custom Product and that Product has been despatched and Invoiced then add the cost to the Sales Order's COGS

Debit	Credit
<i>ASSEMBLY ORDER VARIANCE</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

Inventory Adjustments

The following is used whenever a Stock Adjustment Transaction is carried out through the Inventory Adjustment screen

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>STOCK ADJUSTMENT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or ADJUSTMENTS then the default Cost Centre matched to STOCK or STOCK ADJUSTMENT VARIANCE respectively

will be used

Inventory Count

The following is used whenever a Stock Count Transaction is carried out via the Stock Count routine.

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>STOCK COUNT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or STOCK COUNT then the default Cost Centre matched to STOCK or STOCK COUNT VARIANCE respectively will be used

Inventory Transfer

Inventory Transfer is a two-step function using an interim 'In-Transit' Cost Centre. Currently this process is carried out in a single transaction as follows

Debit	Credit
* (Receiving Warehouse) <i>STOCK</i> <i>STOCK IN TRANSIT</i>	<i>STOCK IN TRANSIT</i> * (Issuing Warehouse) <i>STOCK</i>

* If the Warehouse does not contain a Cost Centre then the INVENTORY Cost Centre then the default Cost Centre matched to STOCK will be used

Negative Stock

For Items that are received where the stock is currently in negative (i.e. has already been issued at a known cost) then the cost difference of the Receipt to the previous issue will be posted to the negative stock adjustment Cost Centre

Debit	Credit
* (Warehouse) <i>STOCK</i>	<i>NEGATIVE ADJUSTMENT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or NEGATIVE STOCK then the default Cost Centre matched to STOCK or NEGATIVE ADJUSTMENT VARIANCE respectively will be used

Inventory Revaluation

Enable any revaluation of Stock to be recorded

Debit	Credit
*(Warehouse) <i>STOCK</i>	<i>REVALUATION OF STOCK VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or RE-VALUATION then the default Cost Centre matched to STOCK or REVALUATION OF STOCK VARIANCE respectively will be used

POS Station Withdrawals/Receipts

Record the Withdrawal or Receipt of miscellaneous cash from the POS Station

Debit	Credit
<i>POS EXPENSES</i>	<i>UNDEPOSITED FUNDS</i>

POS End of Day Variations

Record variation in actual -v- calculated End of Day Cash-up

Debit	Credit
<i>UNDEPOSITED FUNDS</i>	<i>POS END OF DAY</i>

19.1.7 Advanced Cost Centre Mapping

There may be instances where you wish to look deeper in assessing costs within the following areas

- Sales
- Labour
- Inventory

Ostendo will see if the specific record appears here before going to the Base Mapping described in Section 5.

1. Sales Mapping

A Sales Mapping screen extends the base functionality by allowing you to establish a more detailed structure specifically covering the Sales area. Go to **File>Financial Configuration>Sales Mapping Matrix**. This enables you to define Income and COG Cost Centres for a variety of combinations from.

- 1 - Invoice Customer Type
- 2 - Invoice Customer Region
- 3 - Order Customer Type
- 4 - Order Customer Region
- 5 - Order Class
- 6 - Order Type

- 7 - Sales Order Person
- 8 - Category

In this matrix the level 8 (Category) takes the most significant whereas Level 1 (Invoice Customer Type) takes the least significant. For example:

If there are three mapping records containing:

- Category/Order Type/Order Class/Order Customer Type (I.e. Levels 8/6/5/3)
- Category/Sales Order Person/Order Customer Region (I.e. Levels 8/7/4)
- Sales Order Person/Order Class/Invoice Customer Region (I.e. Levels 7/5/2)

If a Sales record had matching fields defined in all three mapping records then the second mapping record would take priority over the others.

If the Sales record had matching fields except for Category defined in all three mapping records then the third mapping record would take priority over the others.

Let's do an example with this Sales Matrix. Say that we wish to segregate Sales by Job Type. The process would be as follows:

Go into **General>Cost Centres** and create one Cost Centre to represent **Income** and another to represent **COGS** for this Job Type

Now go to **Configuration>Sales Mapping Matrix** and click the **'Add'** button to create a new line then go to field **'Order Class'** and select **'Job Orders'** from the drop-down list. Now go to field **'Order Type'** and select **'Progress'**. Finally go to fields 'Income Cost Centre' and 'Cost of Goods Cost Centre' and select the Cost Centres that you have just created.

All Job Order with Order Type **'Progress'** will now be posted to these Cost Centres. Create a Job Order then process the Invoice as described in the Job Orders training Guide then go into **General>Reports** and select **'Financial Batch Reports'**. Enter the following parameters:

- Transaction Status:** Select **'Ready to Send'**
- From Transaction Date:** Select Today's Date
- To Transaction Date:** Select Today's Date
- From Batch No:** Leave Blank
- To Batch No:** Leave Blank
- Exclude Conditions:** 'Check' this checkbox

Click the **OK** button bring the report back to the screen. You will see that the generated Journals now use the Job Order specific Cost Centres

2. Labour Mapping

As with Sales Mapping the Labour Mapping screen extends the base functionality by allowing you to establish a more detailed structure specifically covering Labour activities. Go to **File>Financial Configuration>Labour Mapping Matrix**. This enables you to define Direct Labour, Fixed OH, and Variable O/H Cost Centres for a variety of combinations from.

This screen extends that basic functionality and allows a User to establish a more detailed structure covering the Labour Code area. It consists of a hierarchical structure using

- 1 - Category
- 2 - Labour Code Department
- 3 - Employee Department
- 4 - Labour Code
- 5 - Employee

And mapping them to

- Direct Labour Cost Centre
- Fixed OH Cost Centre
- Variable OH Cost Centre

In the above matrix the level 5 (Employee) takes the most significant whereas Level 1 (Category) takes the least significant. For example:

If there are two mapping records containing:

- Employee/Employee Department/Category (i.e. Levels 5/3/1)
- Employee/Labour Code/Category (i.e. Levels 5/4/1)

The second mapping record would take priority over the first.

3. Inventory Mapping

If you wish to segregate Stock and Stock Movements by Warehouse then you can specifically define Cost Centres to be used by each Warehouse. If the Warehouse is not given any specific Cost centres then the system defaults will apply..

Go to **Inventory>Warehouses** and define Cost centres to cover the following:

Inventory: All Stock in this Warehouse will be posted to this Cost Centre.

Adjustments: Stock Adjustments against this Warehouse are 'received from' or 'issued to' this Cost Centre.

Negative Stock: Stock adjustment variances can occur whenever you receive Stock but the current stock levels are negative. In this case the Average Cost for the Item is not amended but the difference between the Average Cost and the Cost of this receipt is 'posted' to this Cost Centre

Re-Valuation: Whenever the Stock is re-valued (Standard Cost or Average Cost) the difference between the old value and the new value is 'posted' to this Cost Centre

Stock Count: Stock variances (plus or minus) as a result of a Stock Count are 'posted' to this Cost Centre.

Receipt Cost: Used if the Inventory Costing method is 'Standard Costing' and this stores the cost difference between the Assembly Order actual Cost and the Standard Cost of the Item.

19.2 Quickbooks

Ostendo's Accounts Integration function provides facility to have

- Instant integration with Quickbooks using preset Account Codes
- Comprehensive mapping options for more detailed Accounting Structures

The following will be covered in this document:

- Overview
 - Setup steps
 - Process a single transaction from Ostendo to Quickbooks
 - Base Mapping to pre-defined activities
 - 'T' Charts of Journals generated from within Ostendo
 - Additional Sales, Labour and Inventory Mapping
-

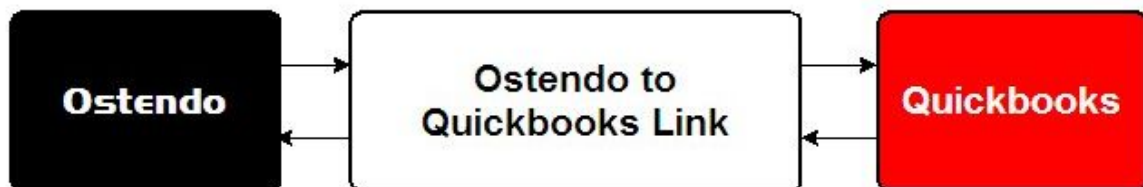
19.2.1 Overview

1. Coverage

Ostendo covers the following functional areas.

- Sales order process comprises Customers, Orders, Delivery, Invoicing, Payment receipts and Banking. Ostendo creates GL Journals and these are posted directly to Quickbooks GL therefore Quickbooks AR is not used
- Purchase Order process comprising Suppliers, Purchase Orders, Goods Receipt, Invoice Receipt and matching. In this process Suppliers are maintained in Ostendo and any new additions or changes in Ostendo will update the Supplier records in Quickbooks. Valid Purchase Order Receipts and Invoices are posted to Quickbooks AP for payment processing
- Assembly Orders are fully maintained in Ostendo and the appropriate Job Costs are recorded and posted to Quickbooks GL
- Inventory issues, receipts and valuations are fully maintained in Ostendo and the appropriate Journals are posted directly to Quickbooks GL

2. Data Flow



Ostendo has a series of Cost Centres against which financial activity is recorded. These are combined to form various Financial Journals covering all activities across Ostendo. A screen is available in Ostendo from which you send a batch of Journals to a separate '[Ostendo to Quickbooks Link](#)' program. It is within this routine that Ostendo's Cost Centres are converted to Quickbooks Account Codes.

The updated results (and possible errors) is returned to the 'Ostendo to Quickbooks Link' from where it can be returned to Ostendo.

19.2.2 Setup

1. Ostendo

Out-of-the-Box Ostendo contains a list of pre-defined areas and their associated Cost Centres. We will expand on these later in this exercise. For now, therefore, no further setup action is required.

The first step is to go to File>System Configuration>Systems Settings and click on the 'Accounting Link' tab. Ensure that the 'Accounting Link Style' is set to 'Quickbooks Link'

2. Quickbooks

Two actions are required in Quickbooks

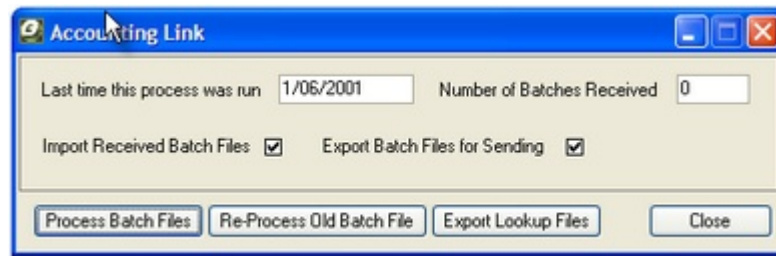
2.1. You should set up a '[Tax Item](#)' in Quickbooks called **OST** with a Tax Rate of Zero and linked to Sales. The reason for this is that both Ostendo and Quickbooks calculate Tax and this could result in minor 'Rounding' variations. Tax Item **OST** allows Development-X to adjust Ostendo's calculated Tax Amount to conform to Quickbooks evaluated amount.

2.2. You should now set up a Customer called **Ostendo** in Quickbooks. Although AR is not used Quickbooks requires a Customer to be included in Journals that originate from AR. Ostendo, therefore will create Journals containing Customer '**Ostendo**'.

3. Ostendo to Quickbooks Link

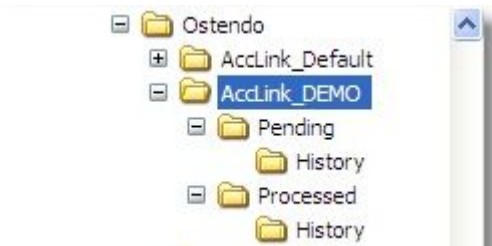
3.1. Export Lookup Files

Before we address the setup in the '**Ostendo to Quickbooks Link**' routine you should first go into Ostendo and go to screen **File>Accounting Link**



On that screen click on the '**Export Lookup Files**' button. This will carry out the following actions:-

3.1.1. Under the Ostendo folder it will create folder **ACCLINC_DEMO** where **DEMO** is the name of the company that you are currently logged into. Under that folder you will see two sub-folders **Pending** and **Processed**. Each of these has a sub-folder called **History**



3.1.2. Directly under the '**Pending**' folder you will see three files

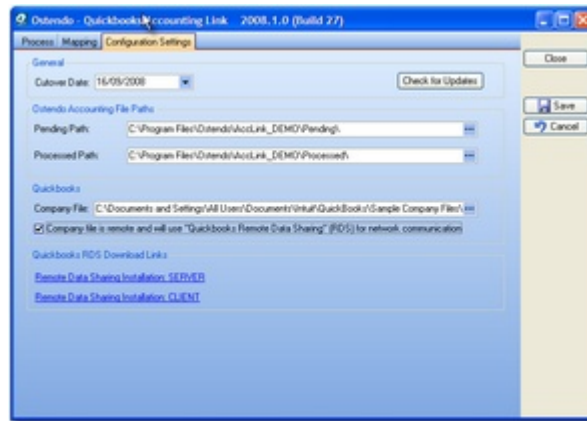
- Costcentre.lup
- Creditterms.lup
- Taxcodes.lup

These 'Lookup Tables' will be referenced during this setup

2. Set Up the Accounting Link

This section describes the fields in the Accounts Interface routine and the activities required. Start up the Ostendo to Accounting Link program and you will be asked to point this routine to both Ostendo and Quickbooks.

Panel 3 - Configuration Settings panel



This panel requires you to point this '**Ostendo to Quickbooks Link**' routine to both Ostendo and Quickbooks. The following fields are shown in this screen

- **Cutover Date:** This is used to allow you to define the date when Ostendo Transactions will begin posting to Quickbooks. Transactions prior to this date will be received by this function but will not be passed onto Quickbooks.

Ostendo Accounting File Paths

- **Receive Path:** This defines where the Batch details received from Ostendo will be stored. Click on the 3 dots icon and select the location of the folder. This will be the **Pending** folder you set up above.
Example **c:\Program Files\Ostendo\Acclinc_DEMO\Pending**
- **Send Path:** This defines where the details of the Batch - after updating Quickbooks - will be stored. Click on the 3 dots icon and select the location of the folder. This will be the **Processed** folder you set up above.
Example **c:\Program Files\Ostendo\Acclinc_DEMO\Processed**

Company File: Click on the 'three dots' Icon and locate the Quickbooks Database

Company File is Remote....: If the above Company file is on a remote machine (Client/Server or Peer to Peer environments) then you should 'check' this checkbox

In this instance Quickbooks requires that a separate application called (RDS) Remote Data Sharing also be installed. This application has two parts

- **Server:** The 'Server' application is to be installed on the Server or PC that has the company file
- **Client:** The Client application is to be installed on the Client

If the checkbox is 'checked' then the following links will be displayed which, when selected, will allow you to download the applications.

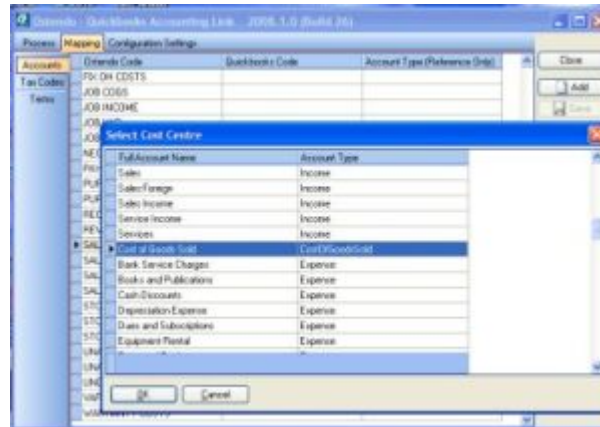
<http://www.development-x.com/downloads/quickbooks/RDSServer.exe>
<http://www.development-x.com/downloads/quickbooks/RDS30Client.exe>

Both these installations have their own help files explaining their use and how to configure them.

Panel 2 - Mapping

This screen requires that you map Ostendo's Cost Centres, Terms Codes and Tax Codes to

Quickbooks's Account Codes, Terms Codes, and Tax Codes.



Step 1. The Ostendo Cost Centres will be displayed in the left-hand column. In the right-hand column select the equivalent Quickbooks Account codes from the drop-down list.

Step 2. Click on the 'Tax Codes' tab down the left side of the screen to display the Tax Codes screen. Click the 'Add' button and select a Tax Code. In the right-hand column select the equivalent Quickbooks Tax Code from the drop-down list. Repeat for each Ostendo Tax Code

Step 3. Click on the 'Terms' tab down the left side of the screen to display the Terms Codes screen. Click the 'Add' button and select a Terms Code. In the right-hand column select the equivalent Quickbooks Terms Code from the drop-down list. Repeat for each Ostendo Terms Code

You have now completed the setup.

Note: Any time you add Cost Centres, Tax Codes or Terms Codes you should go to [File>Accounting Link](#) in Ostendo and download the updated Cost centres, Tax Codes, and Terms Codes then match the new Codes in the Ostendo to Accounting Link program.

19.2.3 Create and post your first Journal

We will create a simple stock adjustment in Ostendo and process that transaction through to Quickbooks. If you go into [Pricing>Item Costing](#) and select Item **100-2002** you will see that it has a current Standard Cost of **\$0.06**. We will receive **100** of these into stock

1. Create a Stock Adjustment

Go into [Inventory>Inventory Adjustments](#) and click the 'Add' button and 'Save' the record. Click on the 'Detail' tab and then the 'Add' button to create a new Line. Enter the following information:

Item Code: **100-2002** (selecting this will prefill the other fields)
Adjustment Type: from the drop-down select 'RECEIPT'
Adjustment Qty (+/-): Enter **100**

Go back to the 'Detail' tab and click on the 'Post all Adjustments' button. This will carry out the adjustment and create the journal. To view the journal go into [General>Reports](#) and select 'Financial Batch Reports'. Enter the following parameters:

Transaction Status: Select 'Ready to Send'

From Transaction Date: Select Today's Date
To Transaction Date: Select Today's Date
From Batch No: Leave Blank
To Batch No: Leave Blank
Exclude Conditions: 'Check' this checkbox

Click the **OK** button bring the report back to the screen

FINANCIAL BATCH REPORT						
Grouped by Status & Type					Development-X Limited	
Transaction Status: Ready to Send						
Transaction Type: Inventory						
Trans Date	Batch No	Debit Cost Centre	Credit Cost Centre	Tax Amt	Tax Code	Amount
24/09/2008		STOCK	STOCK ADJUST	\$0.00		\$6.00
Source: (ADJUSTMENT) Type: RECEIPT Inventory Adjustment: 14						
Totals for: Inventory				\$0.00		\$6.00
Totals for Status: Ready to Send				\$0.00		\$6.00

You will see that the generated journal has a status of '**Ready to Send**' (i.e. ready to send to the [Ostendo to Quickbooks Link](#) program). The journal itself contains the following info that we will focus on

Debits Cost Centre: **STOCK**
 Credits Cost Centre: **STOCKADJUST**

2. Generate a Posting Batch

The next step is to send the created Journal to the '[Ostendo to Quickbooks Link](#)' program.

To do this go into [File>Accounting Link](#) and 'Uncheck' the [Import Received Batch Files](#) checkbox and click the '[Process Batch Files](#)' button. This will gather all Journals with a '**Ready to Send**' status and create a single Batch file. This file will be stored under the '[Pending](#)' folder that you created above.

3. Updating Quickbooks

Go into the [Ostendo to Quickbooks Link](#) program and select the '[Mapping](#)' tab. If you haven't already done so then ensure that a Quickbooks GL Account is associated with Cost Centres **STOCK** and **STOCKADJUST**.

Click on the '[Process](#)' tab where you should see that '**1 file is waiting to be processed**'. Click on '[Process](#)' button and the Link program will update Quickbooks. You will see that progress in the main panel. If any errors are found then they will be displayed here.

You will also find that the Batch file that was in the '[Pending](#)' folder has now moved to [Pending>History](#). Also the Processed Folder will now contain all the processed information about the Batch File

4. Quickbooks

If you now go into Quickbooks you can drill down on the Account Codes or look in the Audit trail where you will see the postings

5. Update Ostendo

The final step in this short exercise is to return the results of the Batch that was sent to the Ostendo to Quickbooks Link program. To do this go into **File>Accounting Link** and make sure that the **Import Received Batch Files** checkbox is 'checked' and click the **'Process Batch Files'** button. This time it will get the Batches currently held in the **'Processed'** folder of the Link program and bring them back into Ostendo. This action will also move the batches to **Processed>History**.

If you now go into **General>Reports** and select **'Financial Batch Reports'**. Enter the following parameters:

- Transaction Status:** Select **'Transaction Valid'**
- From Transaction Date:** Select Today's Date
- To Transaction Date:** Select Today's Date
- From Batch No:** Leave Blank
- To Batch No:** Leave Blank
- Exclude Conditions:** 'Check' this checkbox

Providing the Transaction posted Ok you should see the updated Journal. If the Batch had errors then it should be processed as described below.

That concludes the process flow for a single transaction. However, let us take a closer look at the actions required if any error where found

6. Batches with Errors

6.1. Failed Batch

If there are Batches with errors then the Accounts Link screen will now display an extra button **'Display all Batches that Failed'**.



If you click on this button then all the Batches that have errors will be displayed along with drill-down to show the specific Transactions that contain errors. Action should now be taken to correct the problem(s) that caused the error

Having corrected the errors click on the **'Display All Batches that Failed'** button. Select the Batch Number and then click the **'Re Post Batch'** button. This action will reset those transactions in the Batch whose status is **'Transaction Invalid'** to **'Ready to Send'**

These will be picked up and included in a new batch when you click on the **'Process Batch Files'** button

6.2. Lost Batch

Instances may arise where - for whatever reason - the batch sent to the Ostendo to Quickbooks Link had not been returned with a Batch Status update. Facility is provided where you can resend

this Batch if necessary. If you click on the '[Re-Process Old Batch File](#)' button then a panel will be presented for you to select and re-send the previously sent Batch.

19.2.4 Cost Centres

Ostendo uses the concept of Cost Centres to record costs from all areas of the product. These Cost Centres are user-defined and are directly linked to General Ledger Account Codes in Quickbooks via the [Ostendo to Quickbooks Link](#) program

1. Base Cost Centres

Ostendo comes with 'Base Level' settings in which pre-defined Cost Centres are ready to be mapped to Quickbooks GL Account Codes

Cost Centre Groups

Cost Centres can be grouped into logical areas. These relate to the Cost Areas such as

- Asset
- Liability
- Income
- Expense
- Cost of Sale
- Bank
- etc

Go into [General>Settings>Cost Groups](#) and enter a couple of Groups

Cost Centre Maintenance

If you go into [General>Cost Centres](#) you will see the base set of Cost Centres supplied with Ostendo. In the absence of any changes these Cost Centres will be used when integrating with Quickbooks.

2. Base Cost Centre Mapping

The above Base Cost Centres are linked to areas where costs are stored. Out-of-the-Box Ostendo contains a list of pre-defined areas and their associated Cost Centres. You can see these by going into [File>Financial Configuration>Cost Centre Mapping](#).

The pre-defined areas are:

Stock: All Stock across all Warehouses and Locations in your Company's Inventory is kept against this Cost Centre. If a Warehouse is allocated a specific Cost Centre then that will be used in preference to this default.

Stock Adjustment Variance: Stock Adjustments are 'received from' or 'issued to' a Cost Centre. This Cost Centre provides a global place for 'posting' the Costs from this function

Negative Adjustment Variance: Stock adjustment variances can occur whenever you receive Stock but the current stock levels are negative. In this case the Average Cost for the Item is not amended but the difference between the Average Cost and the Cost of this receipt is 'posted' to this Cost Centre

Revaluation of Stock Variance: Whenever the Stock is re-valued (Standard Cost or Average

Cost) the difference between the old value and the new value is 'posted' to this Cost Centre

Stock Count Variance: Stock variances (plus or minus) as a result of a Stock Count are 'posted' to this Cost Centre.

Stock In Transit: Whenever Stock is moved from one Warehouse to another via the Inventory Transfer routine this Cost Centre will be used for that transfer.

Receipt Cost Var: Used if the Inventory Costing method is 'Standard Costing'. In that instance this stores the cost difference between the Assembly Order actual Cost and the Standard Cost of the Item.

Descriptor Expenses: This is the default Cost Centre used by Descriptors. It can be amended at Descriptor level to another Cost Centre and made specific to that Descriptor. This Cost Centre is then used as the 'allocated' Cost Centre during Purchase Receipts.

Catalogue Expenses: This is the default Cost Centre used by Supplier Catalogue Items. This Cost Centre is used as the 'allocated' Cost Centre during Purchase Receipts. If a Supplier Catalogue is allocated a specific Cost Centre then that will be used in preference to this default.

Warranty Costs: Where a Job Issue has it's 'Actual Issue' allocated to a Charge Style of 'Warranty' then the costs are 'posted' to this Cost Centre

Contract Costs: Where a Job Issue has it's 'Actual Issue' allocated to a Charge Style of 'Contract' then the costs are 'posted' to this Cost Centre

Direct Labour Costs: When an Employee Timesheet Transaction is made then the costs are 'posted' to this Cost Centre

Fixed Labour Overhead Costs: When an Employee Timesheet Transaction is made then the associated FOH costs - held against the linked Labour Code - are 'posted' to this Cost Centre

Variable Labour Overhead Costs: When an Employee Timesheet Transaction is made then the associated VOH costs - held against the linked Labour Code - are 'posted' to this Cost Centre

Purchases Received Not Invoiced: For any Purchase Orders that have been received via the 'Purchase Order Receipts' function are held in this Cost Centre until they have been 'matched' in the 'Purchase Order Invoicing' screen

Purchase Price Variance: Purchase Orders received via the 'Purchase Order Receipts' function contain a Purchase Price and this is 'posted' at that time. When it is matched against the 'Purchase Order Invoicing' there may be a Price difference. This difference is posted to this Cost Centre.

Creditors: Whenever a Purchase Invoice is received then the amount of that Invoice is 'posted' to this Cost Centre.

Assembly Work In Progress: For any Issues (Material, Labour, Descriptors, etc) that have been issued to Work In Progress for an Assembly Order the cost of that issue is 'posted' to this Cost Centre. Note: The cost of the Assembly Order is taken out of this Cost Centre whenever the Assembly Order is received into Inventory

Assembly Order Variance: If the Assembly Order has been completed and its status is 'Closed' then any subsequent Issues or Bookings to this Order will go direct to this Cost Centre rather than the Assembly Work In Progress Cost Centre.

Job Work In Progress: For any Issues (Material, Labour, Descriptors, etc) that have been issued

to Work In Progress for a Job Order the cost of that issue is 'posted' to this Cost Centre. Note: The cost of the Job Order is taken out of this Cost Centre whenever the Job Order is Invoiced.

Job Cost Of Goods Sold: Any Invoice raised against the Job Order will take the cost of the Invoice out of the 'Job Work In Progress' Cost Centre and add it to this Cost Centre.

Job Cost Variance: If the Job Order is 'Closed' then any subsequent Issues or Bookings to this Order will go direct to this Cost Centre rather than the Job Work In Progress Cost Centre.

Job Income: Any Income received that has been applied to a Job Invoice Line will be 'posted' to this Cost Centre.

Sales Lines Picked: Whenever a Sales Order Line is 'Picked' the cost of that line is moved from the source Cost Centre (Example: 'Stock' Cost Centre for Inventory Items) and moved to this Cost Centre.

Sales Cost Of Goods Sold: Whenever a Sales Order Line is 'Invoiced' the cost of that line is moved from the 'Sales Line Picked' Cost Centre to this Cost Centre.

Sales Income: Any Income received that has been applied to a Sales Invoice Line will be 'posted' to this Cost Centre.

Freight: Any Income received that covers Freight and has been applied to an Invoice will be 'posted' to this Cost Centre.

Debtors: Whenever an Invoice (or credit) is raised then the amount of that Invoice is 'posted' to this Cost Centre. Whenever a Deposit or Payments is matched to an Invoice then it is moved from this Cost Centre to the above Sales Invoice, Sales Freight, and Sales tax Cost Centres.

Contract Income: Any Income received that has been applied to a Contract Invoice Line will be 'posted' to this Cost Centre. Until it is 'posted' the income will reside in the 'Unapplied Deposits' or 'Unapplied Payments' Cost Centres

Deferred Cost Of Goods: The value of any Retention amount raised against a Job Order Invoice will take the cost of the Retention out of the 'Job Work In Progress' Cost Centre and add it to this Cost Centre.

Deferred Income: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Income deferred for future P & L recognition.

Deferred Freight: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Freight deferred for future P & L recognition

Deferred Tax: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Tax deferred for future P & L recognition

Un-Applied Deposits: Deposits received from a Customer will be 'posted' to this Cost Centre. When they are applied to an Invoice they will be moved to the 'Sales Income' Cost Centre

Un-Applied Payments: Payments received from a Customer will be 'posted' to this Cost Centre. When they are applied to an Invoice they will be moved to the 'Sales Income' Cost Centre

Bank: Upon validation that the Payment was banked (Deposit Slip status updated to 'Banked' then the amount is transferred from the 'Un-deposited Funds' Cost Centre and 'posted' to this Cost

Centre

Payment Rounding: If the 'matched' payment amount is within a 'Rounding Tolerance' entered in that screen then the Payment record status is amended to 'Fully Applied'. The actual value of the Rounded amount is 'posted' to this Cost Centre,

Discounts Given: If a Discount Amount is entered during Payment Matching (Example: 'Prompt Payment' discount) then the value of this discount is 'posted' to this Cost Centre.

Undeposited Funds: Whenever a Bank Deposit Slip is raised (Status is 'Pending') the included payments are 'posted' to this Cost Centre.

POS End of Day Variance: Whenever a 'Z' Report is printed from the End Of Day processing in the Point of Sales function than if the Till Balance is at variance to the expected balance then the difference is posted to this Cost Centre

POS Shop Expenses: Whenever a Cash withdrawal or deposit is made within the POS function then the amount Withdrawn/Deposited is held in this Cost Centre

19.2.5 Base Mapped Cost Centres

The base mapped Cost Centres in any installation of Ostendo are:

Cost Accumulation Area	Linked Cost Centre
Stock	STOCK
Stock Adjustment Variance	STOCK ADJUST
Negative Stock Variance	NEG STOCK
Revaluation Stock Variance	REVALUE STOCK
Stock Count Variance	COUNT STOCK
Stock In Transit	STOCK INTRANSIT
Receipt Cost Variance	RECEIPT COST VAR
Descriptor Expenses	DESCRIPTOR EXPENSES
Catalogue Expenses	CATALOGUE EXPENSES
Warranty Costs	WARRANTY COSTS
Contract Costs	CONTRACT COSTS
Direct Labour Costs	DIRECT LABOUR COSTS
Fixed Labour Overhead Costs	FIX OH COSTS
Variable Labour Overhead Costs	VAR OH COSTS
Purchases Received Not Invoiced	PURCHASES RECEIPTS
Purchase Price Variance	PURCHASE PRICE VAR

Creditors	CREDITORS
Assembly Work In Progress	ASSEMBLY WIP
Assembly Order Variance	ASSEMBLY VAR
Job Work In Progress	JOB WIP
Job Cost Of Goods Sold	JOB COGS
Job Cost Variance	JOB VAR
Job Income	JOB INCOME
Sales Lines Picked	SALES PICKED
Sales Cost Of Goods Sold	SALES COGS
Sales Income	SALES INCOME
Sales Freight	SALES FREIGHT
Debtors	DEBTORS
Contract Income	CONTRACT INCOME
Deferred Cost Of Goods	DEFERRED COGS
Deferred Income	DEFFERED INCOME
Deferred Freight	DEFFERED FREIGHT
Deferred Tax	DEFFERED TAX
Un-Applied Deposits	UNAPPLIED DEPOSITS
Un-Applied Payments	UNAPPLIED PAYMENTS
Bank	BANK
Payment Rounding	PAYMENT ROUNDING
Discounts Given	DISCOUNTS GIVEN
Un-Deposited Funds	UNDEPOSITED FUNDS
POS End of Day Variance	POS EOD
POS Shop Expenses	POS EXPENSES

19.2.6 'T' Charts

If you go into [General>Reports](#) you will see the following one-page documents showing the flow through that process.

[Financial Flow - Jobs](#)

Financial Flow - Sales
Financial Flow - Assembly
Financial Flow - Purchasing

Each step within those flows use the following 'T' Charts. These are pre-defined in Ostendo and cover the various activities that take place.

Sales Issues

This covers issue of Items, Descriptors and Supplier Catalogue Items to a Sales Order

Debit	Credit
<i>SALES LINES PICKED</i>	*(For Inventory) <i>Cost Centre</i> **(For Descriptors) <i>Cost Centre</i> ***(For Catalogue Items) <i>Cost Centre</i>

* If the Warehouse from where the Item was picked does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Sales Invoice

This covers the production of the Invoice. It should be noted that the Cr values include Tax because Tax is generated in MYOB relating to the Tax Code passed along with the Journal

Debit	Credit
<i>SALES COGS</i> <i>DEBTORS</i>	<i>SALES LINES PICKED</i> <i>SALES INCOME</i> <i>SALES FREIGHT</i>

Purchase Order Receipt

This covers the production of the Invoice. It should be noted that the Cr values include Tax because Tax is generated in MYOB relating to the Tax Code passed along with the Journal

Debit	Credit

*(For Inventory) If using Standard Cost) Cost Centre RECEIPT COST VAR or If not using Standard Cost) Cost Centre **(For Descriptors) Cost Centre *** (For Catalogue Items) Cost Centre (For Direct Allocations) JOB WORK IN PROGRESS or ASSEMBLY WORK IN PROGRESS	PURCHASES RECEIVED NOT INVOICED
--	--

* If the Warehouse into which the Item is received does not contain a Cost Centre then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Purchase Price Variance

This covers situations where the Purchase Order has been received and - at some later time - the Invoice is matched against the Receipt and where the Invoice contains a different price to the receipt price. This difference is posted to a Purchase Price Variance Cost centre

Debit	Credit
(For Inventory) PURCHASE PRICE VARIANCE *(For Descriptors) Cost Centre **(For Catalogue Items) Cost Centre (For Direct Allocations) JOB WORK IN PROGRESS or ASSEMBLY WORK IN PROGRESS	PURCHASES RECEIVED NOT INVOICED

* If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Purchase Order Invoice

The Purchase Invoice is received, matched against prior receipts, and approved.

Debit	Credit
PURCHASES RECEIVED NOT INVOICED	CREDITORS

Job Issues

Issues to Jobs comes from Stock Items, Descriptors, or Labour. These issues could be charged to the Job or can be designated as non-chargeable (covered by Warranty, a Contract, or simply making it non-chargeable when issuing it)

Debit	Credit
JOB WORK IN PROGRESS	*(For Inventory) Cost Centre
(Defined 'Non-Charge' at time of issue) Cost Centre	** (For Descriptors) Cost Centre
(Non-Charge Warranty) WARRANTY COSTS	*** (For Catalogue Items) Cost Centre
(Non-Charge Contract) CONTRACT COSTS	(Labour - broken down into) DIRECT LABOUR COST FIXED LABOUR OVERHEAD COST VARIABLE LABOUR OVERHEAD COST

* If the Warehouse from where the Item was picked does not contain a Cost Centre then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the Cost Centre matched to CATALOGUE EXPENSES will be used

Job Invoice

Upon creation of a Job Invoice 2 Journal transactions are created.

- Relating to the Costs of the Invoice
- Relating to the Expected Income

Debit	Credit
JOB COST OF GOODS SOLD	JOB WORK IN PROGRESS
Debit	Credit
DEBTORS	JOB INCOME * (For Tax) FREIGHT

* Tax is created within MYOB from the Tax Code passed through. I.e. The Job Income amount includes Tax

Job WIP Variance

If a Job's status is changed to 'Closed' then any residual costs for that Job remaining in Work In Progress will be posted to a Job WIP Variance

Debit	Credit
(Invoice Job Style) <i>JOB COST OF GOODS SOLD</i> (If Job Style is 'No Invoice') <i>Cost Centre</i>	<i>JOB WORK IN PROGRESS</i>

Customer Payments

Payments received from a Customer can be deposited directly into the Bank or alternatively retained and 'batched' into a consolidated Bank Deposit Slip.

Debit	Credit
<i>BANK</i> or <i>UNDEPOSITED FUNDS</i>	<i>UNAPPLIED PAYMENTS</i>

Customer Deposits

Deposits received from a Customer can be deposited directly into the Bank or alternatively retained and 'batched' into a consolidated Bank Deposit Slip.

Debit	Credit
<i>BANK</i> or <i>UNDEPOSITED FUNDS</i>	<i>UNAPPLIED DEPOSITS</i>

Applying Payments

Payments are matched against Invoices raised against a Customer

Debit	Credit
<i>UNAPPLIED PAYMENTS</i>	<i>DEBTORS</i> <i>DISCOUNTS GIVEN</i> <i>PAYMENT ROUNDING</i>

Applying Deposits

Deposits are matched against Invoices raised against a Customer

Debit	Credit
<i>UNAPPLIED DEPOSITS</i>	<i>DEBTORS</i> <i>DISCOUNTS GIVEN</i> <i>PAYMENT ROUNDING</i>

Bank Deposits

Bank Deposits are applied Customer Payments and/or Customer Deposits that have been consolidated into a single Bank Deposit Slip

Debit	Credit
<i>BANK</i>	<i>UNDEPOSITED FUNDS</i>

Assembly Issues

Issues to Assembly Orders comes from Stock Items, Descriptors, or Labour.

Debit	Credit
<i>ASSEMBLY WORK IN PROGRESS</i>	*(For Inventory) <i>Cost Centre</i> **(For Descriptors) <i>Cost Centre</i> (Labour - broken down into) <i>DIRECT LABOUR COST</i> <i>FIXED LABOUR OVERHEAD COST</i> <i>VARIABLE LABOUR OVERHEAD COST</i>

* If the Warehouse from where the Item was picked does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

Assembly Receipts

Receipts from Assembly Orders always go into Inventory. The Journal created depends upon whether the Item being received uses Standard Costing or not

With Standard Costing

Debit	Credit
** (Warehouse) <i>Cost Centre</i> * <i>RECEIPT COST VAR</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

* If the Warehouse does not contain a Cost Centre against RECEIPT COST then the default Cost Centre matched to RECEIPT COST VAR will be used. The value of this variance comes from Cost of Receipt - Standard Cost.

No Standard Costing

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

** If the Warehouse does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

Assembly WIP Variance

Whenever an Assembly Order has its status changed to 'Closed' then any residual Costs for the Assembly Order are posted to a WIP Variance. Additional activity is carried out if the following conditions apply

- If the Assembly Order was for a Custom Product and that Product is still in stock then add the Cost to the Stock record
- If the Assembly Order was for a Custom Product and that Product has been withdrawn from stock but has not yet been Invoiced then add the cost to the Sales Order 'Pick but not invoiced' record
- If the Assembly Order was for a Custom Product and that Product has been despatched and Invoiced then add the cost to the Sales Order's COGS

Debit	Credit
<i>ASSEMBLY ORDER VARIANCE</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

Inventory Adjustments

The following is used whenever a Stock Adjustment Transaction is carried out through the Inventory Adjustment screen

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>STOCK ADJUSTMENT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or ADJUSTMENTS then the default Cost Centre matched to STOCK or STOCK ADJUSTMENT VARIANCE respectively will be used

Inventory Count

The following is used whenever a Stock Count Transaction is carried out via the Stock Count routine.

Debit	Credit
*(Warehouse) <i>Cost Centre</i>	<i>STOCK COUNT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or STOCK COUNT then the default Cost Centre matched to STOCK or STOCK COUNT VARIANCE respectively will be used

Inventory Transfer

Inventory Transfer is a two-step function using an interim 'In-Transit' Cost Centre. Currently this process is carried out in a single transaction as follows

Debit	Credit
*(Receiving Warehouse) <i>STOCK</i> <i>STOCK IN TRANSIT</i>	<i>STOCK IN TRANSIT</i> *(Issuing Warehouse) <i>STOCK</i>

* If the Warehouse does not contain a Cost Centre then the INVENTORY Cost Centre then the default Cost Centre matched to STOCK will be used

Negative Stock

For Items that are received where the stock is currently in negative (I.e. has already been issued at a known cost) then the cost difference of the Receipt to the previous issue will be posted to the negative stock adjustment Cost Centre

Debit	Credit
*(Warehouse) <i>STOCK</i>	<i>NEGATIVE ADJUSTMENT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or NEGATIVE STOCK then the default Cost Centre matched to STOCK or NEGATIVE ADJUSTMENT VARIANCE respectively will be used

Inventory Revaluation

Enable any revaluation of Stock to be recorded

Debit	Credit
-------	--------

*(Warehouse) <i>STOCK</i>	<i>REVALUATION OF STOCK VARIANCE</i>
---------------------------	--------------------------------------

* If the Warehouse does not contain a Cost Centre against INVENTORY or RE-VALUATION then the default Cost Centre matched to STOCK or REVALUATION OF STOCK VARIANCE respectively will be used

POS Station Withdrawals/Receipts

Record the Withdrawal or Receipt of miscellaneous cash from the POS Station

Debit	Credit
<i>POS EXPENSES</i>	<i>UNDEPOSITED FUNDS</i>

POS End of Day Variations

Record variation in actual -v- calculated End of Day Cash-up

Debit	Credit
<i>UNDEPOSITED FUNDS</i>	<i>POS END OF DAY</i>

19.2.7 Advanced Cost Centre Mapping

There may be instances where you wish to look deeper in assessing costs within the following areas

- Sales
- Labour
- Inventory

Ostendo will see if the specific record appears here before going to the Base Mapping described in Section 5.

1. Sales Mapping

A Sales Mapping screen extends the base functionality by allowing you to establish a more detailed structure specifically covering the Sales area. Go to **File>Financial Configuration>Sales Mapping Matrix**. This enables you to define Income and COG Cost Centres for a variety of combinations from.

- 1 - Invoice Customer Type
- 2 - Invoice Customer Region
- 3 - Order Customer Type
- 4 - Order Customer Region
- 5 - Order Class
- 6 - Order Type
- 7 - Sales Order Person
- 8 - Category

In this matrix the level 8 (Category) takes the most significant whereas Level 1 (Invoice Customer Type) takes the least significant. For example:

If there are three mapping records containing:

- Category/Order Type/Order Class/Order Customer Type (I.e. Levels 8/6/5/3)
- Category/Sales Order Person/Order Customer Region (I.e. Levels 8/7/4)
- Sales Order Person/Order Class/Invoice Customer Region (I.e. Levels 7/5/2)

If a Sales record had matching fields defined in all three mapping records then the second mapping record would take priority over the others.

If the Sales record had matching fields except for Category defined in all three mapping records then the third mapping record would take priority over the others.

Let's do an example with this Sales Matrix. Say that we wish to segregate Sales by Job Type. The process would be as follows:

Go into **General>Cost Centres** and create one Cost Centre to represent **Income** and another to represent **COGS** for this Job Type

Now go to **Configuration>Sales Mapping Matrix** and click the 'Add' button to create a new line then go to field 'Order Class' and select 'Job Orders' from the drop-down list. Now go to field 'Order Type' and select 'Progress'. Finally go to fields 'Income Cost Centre' and 'Cost of Goods Cost Centre' and select the Cost Centres that you have just created.

All Job Order with Order Type 'Progress' will now be posted to these Cost Centres. Create a Job Order then process the Invoice as described in the Job Orders training Guide then go into **General>Reports** and select 'Financial Batch Reports'. Enter the following parameters:

- Transaction Status:** Select 'Ready to Send'
- From Transaction Date:** Select Today's Date
- To Transaction Date:** Select Today's Date
- From Batch No:** Leave Blank
- To Batch No:** Leave Blank
- Exclude Conditions:** 'Check' this checkbox

Click the **OK** button bring the report back to the screen. You will see that the generated Journals now use the Job Order specific Cost Centres

2. Labour Mapping

As with Sales Mapping the Labour Mapping screen extends the base functionality by allowing you to establish a more detailed structure specifically covering Labour activities. Go to **File>Financial Configuration>Labour Mapping Matrix**. This enables you to define Direct Labour, Fixed OH, and Variable O/H Cost Centres for a variety of combinations from.

This screen extends that basic functionality and allows a User to establish a more detailed structure covering the Labour Code area. It consists of a hierarchical structure using

- 1 - Category
- 2 - Labour Code Department
- 3 - Employee Department
- 4 - Labour Code
- 5 - Employee

And mapping them to

- Direct Labour Cost Centre
- Fixed OH Cost Centre
- Variable OH Cost Centre

In the above matrix the level 5 (Employee) takes the most significant whereas Level 1 (Category) takes the least significant. For example:

If there are two mapping records containing:

- Employee/Employee Department/Category (I.e. Levels 5/3/1)
- Employee/Labour Code/Category (I.e. Levels 5/4/1)

The second mapping record would take priority over the first.

3. Inventory Mapping

If you wish to segregate Stock and Stock Movements by Warehouse then you can specifically define Cost Centres to be used by each Warehouse. If the Warehouse is not given any specific Cost centres then the system defaults will apply..

Go to **Inventory>Warehouses** and define Cost centres to cover the following:

Inventory: All Stock in this Warehouse will be posted to this Cost Centre.

Adjustments: Stock Adjustments against this Warehouse are 'received from' or 'issued to' this Cost Centre.

Negative Stock: Stock adjustment variances can occur whenever you receive Stock but the current stock levels are negative. In this case the Average Cost for the Item is not amended but the difference between the Average Cost and the Cost of this receipt is 'posted' to this Cost Centre

Re-Valuation: Whenever the Stock is re-valued (Standard Cost or Average Cost) the difference between the old value and the new value is 'posted' to this Cost Centre

Stock Count: Stock variances (plus or minus) as a result of a Stock Count are 'posted' to this Cost Centre.

Receipt Cost: Used if the Inventory Costing method is 'Standard Costing' and this stores the cost difference between the Assembly Order actual Cost and the Standard Cost of the Item.

19.3 Sage Pastel Evolution

Ostendo's Accounts Integration function provides facility to have

- Instant integration with Sage Pastel Evolution using preset Account Codes
- Comprehensive mapping options for more detailed Accounting Structures

The following will be covered in this document:

- Overview
- Setup steps
- Process a single transaction from Ostendo to Sage Pastel Evolution
- Base Mapping to pre-defined activities
- 'T' Charts of Journals generated from within Ostendo
- Additional Sales, Labour and Inventory Mapping

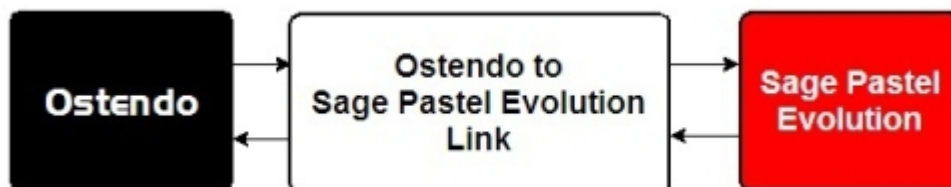
19.3.1 Overview

1. Coverage

Ostendo covers the following functional areas.

- Sales order process comprises Customers, Orders, Delivery, Invoicing, Payment receipts and Banking. Ostendo creates GL Journals and these are posted directly to Sage Pastel Evolution GL therefore Pastel Evolution AR is not used
- Purchase Order process comprising Suppliers, Purchase Orders, Goods Receipt, Invoice Receipt and matching. In this process Suppliers are maintained in Ostendo and any new additions or changes in Ostendo will update the Supplier records in Sage Pastel Evolution. Valid Purchase Order Receipts and Invoices are posted to Sage Pastel AP for payment processing
- Assembly Orders are fully maintained in Ostendo and the appropriate Job Costs are recorded and posted to Sage Pastel Evolution GL
- Inventory issues, receipts and valuations are fully maintained in Ostendo and the appropriate Journals are posted directly to Sage Pastel Evolution GL

2. Data Flow



Ostendo has a series of Cost Centres against which financial activity is recorded. These are combined to form various Financial Journals covering all activities across Ostendo. A screen is available in Ostendo from which you send a batch of Journals to a separate '[Ostendo to Sage Pastel Evolution Link](#)' program. It is within this routine that Ostendo's Cost Centres are converted to Sage Pastel Evolution Account Codes.

The updated results (and possible errors) is returned to the 'Ostendo to Sage Pastel Evolution Link' from where it can be returned to Ostendo.

19.3.2 Setup

1. Ostendo

Out-of-the-Box Ostendo contains a list of pre-defined areas and their associated Cost Centres. We will expand on these later in this exercise. For now, therefore, no further setup action is required.

2. Ostendo setup

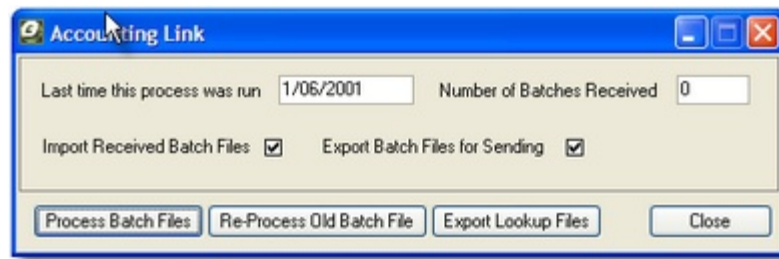
Go to File>System Configuration>Systems Settings and click on the 'Accounting Link' tab. Ensure that the 'Accounting Link Style' is set to 'Sage Pastel Evolution'

3. Ostendo to Sage Pastel Evolution Link

3.1. Export Lookup Files

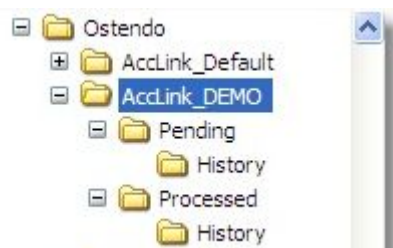
Before we address the setup in the '[Ostendo to Sage Pastel Evolution Link](#)' routine you should

first go into Ostendo and go to screen **File>Accounting Link**



On that screen click on the '**Export Lookup Files**' button. This will carry out the following actions:-

3.1.1. Under the Ostendo folder it will create folder **ACCLINC_DEMO** where **DEMO** is the name of the company that you are currently logged into. Under that folder you will see two sub-folders **Pending** and **Processed**. Each of these has a sub-folder called **History**



3.1.2. Directly under the '**Pending**' folder you will see three files

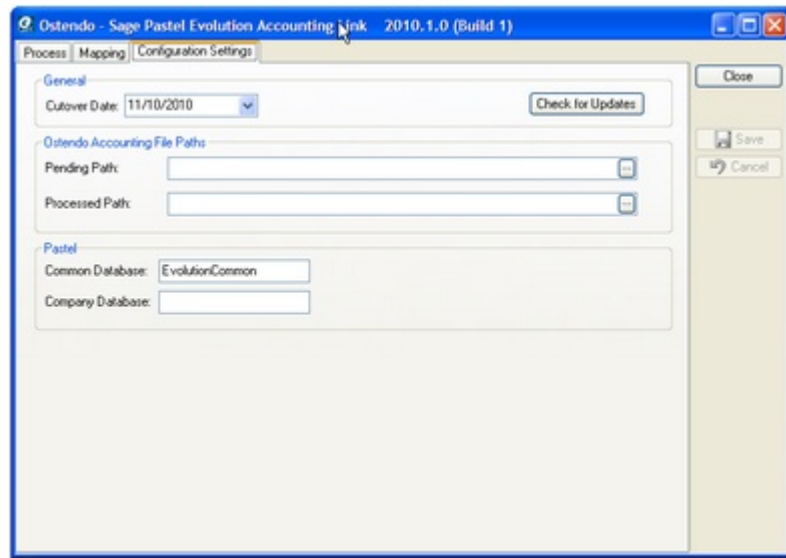
- Costcentre.lup
- Creditterms.lup
- Taxcodes.lup

These 'Lookup Tables' will be referenced during this setup

2. Set Up the Accounting Link

This section describes the fields in the Accounts Interface routine and the activities required. Start up the Ostendo to Accounting Link program and you will be asked to point this routine to both Ostendo and Pastel Evolution.

Panel 3 - Configuration Settings panel



This panel requires you to point this '**Ostendo to Sage Pastel Evolution Link**' routine to both Ostendo and Evolution. The following fields are shown in this screen

- **Cutover Date:** This is used to allow you to define the date when Ostendo Transactions will begin posting to Pastel Evolution. Transactions prior to this date will be received by this function but will not be passed onto Pastel Evolution.

Ostendo Accounting File Paths

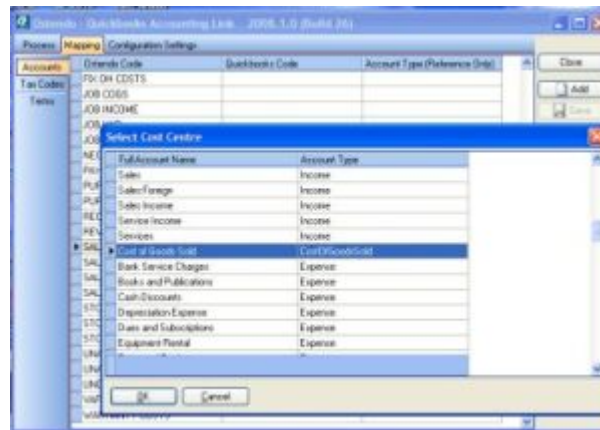
- **Pending Path:** This defines where the Batch details received from Ostendo will be stored. Click on the 3 dots icon and select the location of the folder. This will be the **Pending** folder you set up above.
Example **c:\Program Files\Ostendo\Acclinc_DEMOPending**
- **Processed Path:** This defines where the details of the Batch - after updating Pastel Evolution - will be stored. Click on the 3 dots icon and select the location of the folder. This will be the **Processed** folder you set up above.
Example **c:\Program Files\Ostendo\Acclinc_DEMOProcessed**

Common Database: Enter the name of the Pastel Evolution common Database

Company Database: Enter the name of the Pastel Evolution Company Database

Panel 2 - Mapping

This screen requires that you map Ostendo's Cost Centres, Terms Codes and Tax Codes to Pastel Evolution's Account Codes, Terms Codes, and Tax Codes.



Step 1. The Ostendo Cost Centres will be displayed in the left-hand column. In the right-hand column select the equivalent Pastel Evolution Account codes from the drop-down list.

Step 2. Click on the **'Tax Codes'** tab down the left side of the screen to display the Tax Codes screen. Click the **'Add'** button and select a Tax Code. In the right-hand column select the equivalent Pastel Evolution Tax Code from the drop-down list. Repeat for each Ostendo Tax Code

Step 3. Click on the **'Terms'** tab down the left side of the screen to display the Terms Codes screen. Click the **'Add'** button and select a Terms Code. In the right-hand column select the equivalent Pastel Evolution Terms Code from the drop-down list. Repeat for each Ostendo Terms Code

You have now completed the setup.

Note: Any time you add Cost Centres, Tax Codes or Terms Codes you should go to **File>Accounting Link** in Ostendo and download the updated Cost centres, Tax Codes, and Terms Codes then match the new Codes in the Ostendo to Accounting Link program.

19.3.3 Create and post your first Journal

We will create a simple stock adjustment in Ostendo and process that transaction through to Pastel. If you go into **Pricing>Item Costing** and select Item **100-2002** you will see that it has a current Standard Cost of **\$0.06**. We will receive **100** of these into stock

1. Create a Stock Adjustment

Go into **Inventory>Inventory Adjustments** and click the **'Add'** button and **'Save'** the record. Click on the **'Detail'** tab and then the **'Add'** button to create a new Line. Enter the following information:

Item Code: **100-2002** (selecting this will prefill the other fields)
Adjustment Type: from the drop-down select **'RECEIPT'**
Adjustment Qty (+/-): Enter **100**

Go back to the **'Detail'** tab and click on the **'Post all Adjustments'** button. This will carry out the adjustment and create the journal. To view the journal go into **General>Reports** and select **'Financial Batch Reports'**. Enter the following parameters:

Transaction Status: Select **'Ready to Send'**
From Transaction Date: Select Today's Date
To Transaction Date: Select Today's Date

From Batch No: Leave Blank
To Batch No: Leave Blank
Exclude Conditions: 'Check' this checkbox

Click the **OK** button bring the report back to the screen

FINANCIAL BATCH REPORT						
Grouped by Status & Type						Development-X Limited
Transaction Status: Ready to Send						
Transaction Type: Inventory						
Trans Date	Batch No	Debit Cost Centre	Credit Cost Centre	Tax Amt	Tax Code	Amount
24/09/2008		STOCK	STOCK ADJUST	\$0.00		\$6.00
Source:(ADJUSTMENT) Type:RECEPT) Inventory Adjustment: 14						
Totals for: Inventory				\$0.00		\$6.00
Totals for Status: Ready to Send				\$0.00		\$6.00

You will see that the generated journal has a status of '**Ready to Send**' (i.e. ready to send to the **Ostendo to Sage Pastel Evolution Link** program. The journal itself contains the following info that we will focus on

Debits Cost Centre: **STOCK**
Credits Cost Centre: **STOCKADJUST**

2. Generate a Posting Batch

The next step is to send the created Journal to the '**Ostendo to Sage Pastel Evolution Link**' program.

To do this go into **File>Accounting Link** and 'Uncheck' the **Import Received Batch Files** checkbox and click the '**Process Batch Files**' button. This will gather all Journals with a '**Ready to Send**' status and create a single Batch file. This file will be stored under the '**Pending**' folder that you created above.

3. Updating Pastel Evolution

Go into the **Ostendo to Sage Pastel Evolution Link** program and select the '**Mapping**' tab. If you haven't already done so then ensure that a Pastel Evolution GL Account is associated with Cost Centres **STOCK** and **STOCKADJUST**.

Click on the '**Process**' tab where you should see that '**1 file is waiting to be processed**'. Click on '**Process**' button and the Link program will update Quickbooks. You will see that progress in the main panel. If any errors are found then they will be displayed here.

You will also find that the Batch file that was in the '**Pending**' folder has now moved to **Pending>History**. Also the Processed Folder will now contain all the processed information about the Batch File

4. Pastel Evolution

If you now go into Pastel Evolution you can drill down on the Account Codes or look in the Audit trail where you will see the postings

5. Update Ostendo

The final step in this short exercise is to return the results of the Batch that was sent to the *Ostendo to Sage Pastel Evolution Link* program. To do this go into *File>Accounting Link* and make sure that the *Import Received Batch Files* checkbox is 'checked' and click the *'Process Batch Files'* button. This time it will get the Batches currently held in the *'Processed'* folder of the Link program and bring them back into Ostendo. This action will also move the batches to *Processed>History*.

If you now go into *General>Reports* and select *'Financial Batch Reports'*. Enter the following parameters:

- Transaction Status:** Select **'Transaction Valid'**
- From Transaction Date:** Select Today's Date
- To Transaction Date:** Select Today's Date
- From Batch No:** Leave Blank
- To Batch No:** Leave Blank
- Exclude Conditions:** 'Check' this checkbox

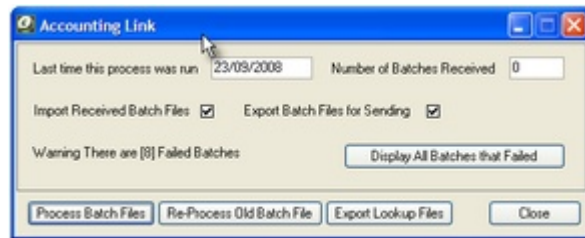
Providing the Transaction posted Ok you should see the updated Journal. If the Batch had errors then it should be processed as described below.

That concludes the process flow for a single transaction. However, let us take a closer look at the actions required if any error where found

6. Batches with Errors

6.1. Failed Batch

If there are Batches with errors then the Accounts Link screen will now display an extra button *'Display all Batches that Failed'*.



If you click on this button then all the Batches that have errors will be displayed along with drill-down to show the specific Transactions that contain errors. Action should now be taken to correct the problem(s) that caused the error

Having corrected the errors click on the *'Display All Batches that Failed'* button. Select the Batch Number and then click the *'Re Post Batch'* button. This action will reset those transactions in the Batch whose status is **'Transaction Invalid'** to **'Ready to Send'**

These will be picked up and included in a new batch when you click on the *'Process Batch Files'* button

6.2. Lost Batch

Instances may arise where - for whatever reason - the batch sent to the Ostendo to Pastel Evolution Link had not been returned with a Batch Status update. Facility is provided where you

can resend this Batch if necessary. If you click on the '[Re-Process Old Batch File](#)' button then a panel will be presented for you to select and re-send the previously sent Batch.

19.3.4 Cost Centres

Ostendo uses the concept of Cost Centres to record costs from all areas of the product. These Cost Centres are user-defined and are directly linked to General Ledger Account Codes in Pastel Evolution via the [Ostendo to Sage Pastel Evolution Link](#) program

1. Base Cost Centres

Ostendo comes with 'Base Level' settings in which pre-defined Cost Centres are ready to be mapped to Pastel Evolution GL Account Codes

Cost Centre Groups

Cost Centres can be grouped into logical areas. These relate to the Cost Areas such as

- Asset
- Liability
- Income
- Expense
- Cost of Sale
- Bank
- etc

Go into [General>Settings>Cost Groups](#) and enter a couple of Groups

Cost Centre Maintenance

If you go into [General>Cost Centres](#) you will see the base set of Cost Centres supplied with Ostendo. In the absence of any changes these Cost Centres will be used when integrating with Pastel Evolution.

2. Base Cost Centre Mapping

The above Base Cost Centres are linked to areas where costs are stored. Out-of-the-Box Ostendo contains a list of pre-defined areas and their associated Cost Centres. You can see these by going into [File>Financial Configuration>Cost Centre Mapping](#).

The pre-defined areas are:

Stock: All Stock across all Warehouses and Locations in your Company's Inventory is kept against this Cost Centre. If a Warehouse is allocated a specific Cost Centre then that will be used in preference to this default.

Stock Adjustment Variance: Stock Adjustments are 'received from' or 'issued to' a Cost Centre. This Cost Centre provides a global place for 'posting' the Costs from this function

Negative Adjustment Variance: Stock adjustment variances can occur whenever you receive Stock but the current stock levels are negative. In this case the Average Cost for the Item is not amended but the difference between the Average Cost and the Cost of this receipt is 'posted' to this Cost Centre

Revaluation of Stock Variance: Whenever the Stock is re-valued (Standard Cost or Average

Cost) the difference between the old value and the new value is 'posted' to this Cost Centre

Stock Count Variance: Stock variances (plus or minus) as a result of a Stock Count are 'posted' to this Cost Centre.

Stock In Transit: Whenever Stock is moved from one Warehouse to another via the Inventory Transfer routine this Cost Centre will be used for that transfer.

Receipt Cost Var: Used if the Inventory Costing method is 'Standard Costing'. In that instance this stores the cost difference between the Assembly Order actual Cost and the Standard Cost of the Item.

Descriptor Expenses: This is the default Cost Centre used by Descriptors. It can be amended at Descriptor level to another Cost Centre and made specific to that Descriptor. This Cost Centre is then used as the 'allocated' Cost Centre during Purchase Receipts.

Catalogue Expenses: This is the default Cost Centre used by Supplier Catalogue Items. This Cost Centre is used as the 'allocated' Cost Centre during Purchase Receipts. If a Supplier Catalogue is allocated a specific Cost Centre then that will be used in preference to this default.

Warranty Costs: Where a Job Issue has it's 'Actual Issue' allocated to a Charge Style of 'Warranty' then the costs are 'posted' to this Cost Centre

Contract Costs: Where a Job Issue has it's 'Actual Issue' allocated to a Charge Style of 'Contract' then the costs are 'posted' to this Cost Centre

Direct Labour Costs: When an Employee Timesheet Transaction is made then the costs are 'posted' to this Cost Centre

Fixed Labour Overhead Costs: When an Employee Timesheet Transaction is made then the associated FOH costs - held against the linked Labour Code - are 'posted' to this Cost Centre

Variable Labour Overhead Costs: When an Employee Timesheet Transaction is made then the associated VOH costs - held against the linked Labour Code - are 'posted' to this Cost Centre

Purchases Received Not Invoiced: For any Purchase Orders that have been received via the 'Purchase Order Receipts' function are held in this Cost Centre until they have been 'matched' in the 'Purchase Order Invoicing' screen

Purchase Price Variance: Purchase Orders received via the 'Purchase Order Receipts' function contain a Purchase Price and this is 'posted' at that time. When it is matched against the 'Purchase Order Invoicing' there may be a Price difference. This difference is posted to this Cost Centre.

Creditors: Whenever a Purchase Invoice is received then the amount of that Invoice is 'posted' to this Cost Centre.

Assembly Work In Progress: For any Issues (Material, Labour, Descriptors, etc) that have been issued to Work In Progress for an Assembly Order the cost of that issue is 'posted' to this Cost Centre. Note: The cost of the Assembly Order is taken out of this Cost Centre whenever the Assembly Order is received into Inventory

Assembly Order Variance: If the Assembly Order has been completed and its status is 'Closed' then any subsequent Issues or Bookings to this Order will go direct to this Cost Centre rather than the Assembly Work In Progress Cost Centre.

Job Work In Progress: For any Issues (Material, Labour, Descriptors, etc) that have been issued

to Work In Progress for a Job Order the cost of that issue is 'posted' to this Cost Centre. Note: The cost of the Job Order is taken out of this Cost Centre whenever the Job Order is Invoiced.

Job Cost Of Goods Sold: Any Invoice raised against the Job Order will take the cost of the Invoice out of the 'Job Work In Progress' Cost Centre and add it to this Cost Centre.

Job Cost Variance: If the Job Order is 'Closed' then any subsequent Issues or Bookings to this Order will go direct to this Cost Centre rather than the Job Work In Progress Cost Centre.

Job Income: Any Income received that has been applied to a Job Invoice Line will be 'posted' to this Cost Centre.

Sales Lines Picked: Whenever a Sales Order Line is 'Picked' the cost of that line is moved from the source Cost Centre (Example: 'Stock' Cost Centre for Inventory Items) and moved to this Cost Centre.

Sales Cost Of Goods Sold: Whenever a Sales Order Line is 'Invoiced' the cost of that line is moved from the 'Sales Line Picked' Cost Centre to this Cost Centre.

Sales Income: Any Income received that has been applied to a Sales Invoice Line will be 'posted' to this Cost Centre.

Freight: Any Income received that covers Freight and has been applied to an Invoice will be 'posted' to this Cost Centre.

Debtors: Whenever an Invoice (or credit) is raised then the amount of that Invoice is 'posted' to this Cost Centre. Whenever a Deposit or Payments is matched to an Invoice then it is moved from this Cost Centre to the above Sales Invoice, Sales Freight, and Sales tax Cost Centres.

Contract Income: Any Income received that has been applied to a Contract Invoice Line will be 'posted' to this Cost Centre. Until it is 'posted' the income will reside in the 'Unapplied Deposits' or 'Unapplied Payments' Cost Centres

Deferred Cost Of Goods: The value of any Retention amount raised against a Job Order Invoice will take the cost of the Retention out of the 'Job Work In Progress' Cost Centre and add it to this Cost Centre.

Deferred Income: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Income deferred for future P & L recognition.

Deferred Freight: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Freight deferred for future P & L recognition

Deferred Tax: This is only used in 'Progress Claim' environments that have an Invoicing Schedule where the specific scheduled Invoice has an Accounting Style of 'Income Deferred'. This provides facility to have this type of Tax deferred for future P & L recognition

Un-Applied Deposits: Deposits received from a Customer will be 'posted' to this Cost Centre. When they are applied to an Invoice they will be moved to the 'Sales Income' Cost Centre

Un-Applied Payments: Payments received from a Customer will be 'posted' to this Cost Centre. When they are applied to an Invoice they will be moved to the 'Sales Income' Cost Centre

Bank: Upon validation that the Payment was banked (Deposit Slip status updated to 'Banked' then the amount is transferred from the 'Un-deposited Funds' Cost Centre and 'posted' to this Cost

Centre

Payment Rounding: If the 'matched' payment amount is within a 'Rounding Tolerance' entered in that screen then the Payment record status is amended to 'Fully Applied'. The actual value of the Rounded amount is 'posted' to this Cost Centre,

Discounts Given: If a Discount Amount is entered during Payment Matching (Example: 'Prompt Payment' discount) then the value of this discount is 'posted' to this Cost Centre.

Undeposited Funds: Whenever a Bank Deposit Slip is raised (Status is 'Pending') the included payments are 'posted' to this Cost Centre.

POS End of Day Variance: Whenever a 'Z' Report is printed from the End Of Day processing in the Point of Sales function than if the Till Balance is at variance to the expected balance then the difference is posted to this Cost Centre

POS Shop Expenses: Whenever a Cash withdrawal or deposit is made within the POS function then the amount Withdrawn/Deposited is held in this Cost Centre

19.3.5 Base Mapped Cost Centres

The base mapped Cost Centres in any installation of Ostendo are:

Cost Accumulation Area	Linked Cost Centre
Stock	STOCK
Stock Adjustment Variance	STOCK ADJUST
Negative Stock Variance	NEG STOCK
Revaluation Stock Variance	REVALUE STOCK
Stock Count Variance	COUNT STOCK
Stock In Transit	STOCK INTRANSIT
Receipt Cost Variance	RECEIPT COST VAR
Descriptor Expenses	DESCRIPTOR EXPENSES
Catalogue Expenses	CATALOGUE EXPENSES
Warranty Costs	WARRANTY COSTS
Contract Costs	CONTRACT COSTS
Direct Labour Costs	DIRECT LABOUR COSTS
Fixed Labour Overhead Costs	FIX OH COSTS
Variable Labour Overhead Costs	VAR OH COSTS
Purchases Received Not Invoiced	PURCHASES RECEIPTS
Purchase Price Variance	PURCHASE PRICE VAR

Creditors	CREDITORS
Assembly Work In Progress	ASSEMBLY WIP
Assembly Order Variance	ASSEMBLY VAR
Job Work In Progress	JOB WIP
Job Cost Of Goods Sold	JOB COGS
Job Cost Variance	JOB VAR
Job Income	JOB INCOME
Sales Lines Picked	SALES PICKED
Sales Cost Of Goods Sold	SALES COGS
Sales Income	SALES INCOME
Sales Freight	SALES FREIGHT
Debtors	DEBTORS
Contract Income	CONTRACT INCOME
Deferred Cost Of Goods	DEFERRED COGS
Deferred Income	DEFFERED INCOME
Deferred Freight	DEFFERED FREIGHT
Deferred Tax	DEFFERED TAX
Un-Applied Deposits	UNAPPLIED DEPOSITS
Un-Applied Payments	UNAPPLIED PAYMENTS
Bank	BANK
Payment Rounding	PAYMENT ROUNDING
Discounts Given	DISCOUNTS GIVEN
Un-Deposited Funds	UNDEPOSITED FUNDS
POS End of Day Variance	POS EOD
POS Shop Expenses	POS EXPENSES

19.3.6 'T' Charts

If you go into [General>Reports](#) you will see the following one-page documents showing the flow through that process.

[Financial Flow - Jobs](#)

Financial Flow - Sales
Financial Flow - Assembly
Financial Flow - Purchasing

Each step within those flows use the following 'T' Charts. These are pre-defined in Ostendo and cover the various activities that take place.

Sales Issues

This covers issue of Items, Descriptors and Supplier Catalogue Items to a Sales Order

Debit	Credit
<i>SALES LINES PICKED</i>	*(For Inventory) <i>Cost Centre</i> **(For Descriptors) <i>Cost Centre</i> ***(For Catalogue Items) <i>Cost Centre</i>

* If the Warehouse from where the Item was picked does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Sales Invoice

This covers the production of the Invoice. It should be noted that the Cr values include Tax because Tax is generated in MYOB relating to the Tax Code passed along with the Journal

Debit	Credit
<i>SALES COGS</i> <i>DEBTORS</i>	<i>SALES LINES PICKED</i> <i>SALES INCOME</i> <i>SALES FREIGHT</i>

Purchase Order Receipt

This covers the production of the Invoice. It should be noted that the Cr values include Tax because Tax is generated in MYOB relating to the Tax Code passed along with the Journal

Debit	Credit

*(For Inventory) If using Standard Cost) Cost Centre RECEIPT COST VAR or If not using Standard Cost) Cost Centre **(For Descriptors) Cost Centre *** (For Catalogue Items) Cost Centre (For Direct Allocations) JOB WORK IN PROGRESS or ASSEMBLY WORK IN PROGRESS	PURCHASES RECEIVED NOT INVOICED
--	--

* If the Warehouse into which the Item is received does not contain a Cost Centre then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Purchase Price Variance

This covers situations where the Purchase Order has been received and - at some later time - the Invoice is matched against the Receipt and where the Invoice contains a different price to the receipt price. This difference is posted to a Purchase Price Variance Cost centre

Debit	Credit
(For Inventory) PURCHASE PRICE VARIANCE *(For Descriptors) Cost Centre **(For Catalogue Items) Cost Centre (For Direct Allocations) JOB WORK IN PROGRESS or ASSEMBLY WORK IN PROGRESS	PURCHASES RECEIVED NOT INVOICED

* If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the default Cost Centre matched to CATALOGUE EXPENSES will be used

Purchase Order Invoice

The Purchase Invoice is received, matched against prior receipts, and approved.

Debit	Credit
PURCHASES RECEIVED NOT INVOICED	CREDITORS

Job Issues

Issues to Jobs comes from Stock Items, Descriptors, or Labour. These issues could be charged to the Job or can be designated as non-chargeable (covered by Warranty, a Contract, or simply making it non-chargeable when issuing it)

Debit	Credit
JOB WORK IN PROGRESS	*(For Inventory) Cost Centre
(Defined 'Non-Charge' at time of issue) Cost Centre	** (For Descriptors) Cost Centre
(Non-Charge Warranty) WARRANTY COSTS	*** (For Catalogue Items) Cost Centre
(Non-Charge Contract) CONTRACT COSTS	(Labour - broken down into) DIRECT LABOUR COST FIXED LABOUR OVERHEAD COST VARIABLE LABOUR OVERHEAD COST

* If the Warehouse from where the Item was picked does not contain a Cost Centre then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the Cost Centre matched to DESCRIPTOR EXPENSES will be used

*** If the Supplier Catalogue from where the Item is issued does not contain a Cost Centre then the Cost Centre matched to CATALOGUE EXPENSES will be used

Job Invoice

Upon creation of a Job Invoice 2 Journal transactions are created.

- Relating to the Costs of the Invoice
- Relating to the Expected Income

Debit	Credit
JOB COST OF GOODS SOLD	JOB WORK IN PROGRESS
Debit	Credit
DEBTORS	JOB INCOME * (For Tax) FREIGHT

* Tax is created within MYOB from the Tax Code passed through. I.e. The Job Income amount includes Tax

Job WIP Variance

If a Job's status is changed to 'Closed' then any residual costs for that Job remaining in Work In Progress will be posted to a Job WIP Variance

Debit	Credit
(Invoice Job Style) JOB COST OF GOODS SOLD (If Job Style is 'No Invoice') Cost Centre	JOB WORK IN PROGRESS

Customer Payments

Payments received from a Customer can be deposited directly into the Bank or alternatively retained and 'batched' into a consolidated Bank Deposit Slip.

Debit	Credit
BANK or UNDEPOSITED FUNDS	UNAPPLIED PAYMENTS

Customer Deposits

Deposits received from a Customer can be deposited directly into the Bank or alternatively retained and 'batched' into a consolidated Bank Deposit Slip.

Debit	Credit
BANK or UNDEPOSITED FUNDS	UNAPPLIED DEPOSITS

Applying Payments

Payments are matched against Invoices raised against a Customer

Debit	Credit
UNAPPLIED PAYMENTS	DEBTORS DISCOUNTS GIVEN PAYMENT ROUNDING

Applying Deposits

Deposits are matched against Invoices raised against a Customer

Debit	Credit
<i>UNAPPLIED DEPOSITS</i>	<i>DEBTORS</i> <i>DISCOUNTS GIVEN</i> <i>PAYMENT ROUNDING</i>

Bank Deposits

Bank Deposits are applied Customer Payments and/or Customer Deposits that have been consolidated into a single Bank Deposit Slip

Debit	Credit
<i>BANK</i>	<i>UNDEPOSITED FUNDS</i>

Assembly Issues

Issues to Assembly Orders comes from Stock Items, Descriptors, or Labour.

Debit	Credit
<i>ASSEMBLY WORK IN PROGRESS</i>	*(For Inventory) <i>Cost Centre</i> **(For Descriptors) <i>Cost Centre</i> (Labour - broken down into) <i>DIRECT LABOUR COST</i> <i>FIXED LABOUR OVERHEAD COST</i> <i>VARIABLE LABOUR OVERHEAD COST</i>

* If the Warehouse from where the Item was picked does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

** If the Descriptor does not contain a Cost Centre then the default Cost Centre matched to DESCRIPTOR EXPENSES will be used

Assembly Receipts

Receipts from Assembly Orders always go into Inventory. The Journal created depends upon whether the Item being received uses Standard Costing or not

With Standard Costing

Debit	Credit
** (Warehouse) <i>Cost Centre</i> * <i>RECEIPT COST VAR</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

* If the Warehouse does not contain a Cost Centre against RECEIPT COST then the default Cost Centre matched to RECEIPT COST VAR will be used. The value of this variance comes from Cost of Receipt - Standard Cost.

No Standard Costing

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

** If the Warehouse does not contain a Cost Centre against INVENTORY then the default Cost Centre matched to STOCK will be used

Assembly WIP Variance

Whenever an Assembly Order has its status changed to 'Closed' then any residual Costs for the Assembly Order are posted to a WIP Variance. Additional activity is carried out if the following conditions apply

- If the Assembly Order was for a Custom Product and that Product is still in stock then add the Cost to the Stock record
- If the Assembly Order was for a Custom Product and that Product has been withdrawn from stock but has not yet been Invoiced then add the cost to the Sales Order 'Pick but not invoiced' record
- If the Assembly Order was for a Custom Product and that Product has been despatched and Invoiced then add the cost to the Sales Order's COGS

Debit	Credit
<i>ASSEMBLY ORDER VARIANCE</i>	<i>ASSEMBLY WORK IN PROGRESS</i>

Inventory Adjustments

The following is used whenever a Stock Adjustment Transaction is carried out through the Inventory Adjustment screen

Debit	Credit
* (Warehouse) <i>Cost Centre</i>	<i>STOCK ADJUSTMENT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or ADJUSTMENTS then the default Cost Centre matched to STOCK or STOCK ADJUSTMENT VARIANCE respectively will be used

Inventory Count

The following is used whenever a Stock Count Transaction is carried out via the Stock Count routine.

Debit	Credit
*(Warehouse) <i>Cost Centre</i>	<i>STOCK COUNT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or STOCK COUNT then the default Cost Centre matched to STOCK or STOCK COUNT VARIANCE respectively will be used

Inventory Transfer

Inventory Transfer is a two-step function using an interim 'In-Transit' Cost Centre. Currently this process is carried out in a single transaction as follows

Debit	Credit
*(Receiving Warehouse) <i>STOCK</i> <i>STOCK IN TRANSIT</i>	<i>STOCK IN TRANSIT</i> *(Issuing Warehouse) <i>STOCK</i>

* If the Warehouse does not contain a Cost Centre then the INVENTORY Cost Centre then the default Cost Centre matched to STOCK will be used

Negative Stock

For Items that are received where the stock is currently in negative (I.e. has already been issued at a known cost) then the cost difference of the Receipt to the previous issue will be posted to the negative stock adjustment Cost Centre

Debit	Credit
*(Warehouse) <i>STOCK</i>	<i>NEGATIVE ADJUSTMENT VARIANCE</i>

* If the Warehouse does not contain a Cost Centre against INVENTORY or NEGATIVE STOCK then the default Cost Centre matched to STOCK or NEGATIVE ADJUSTMENT VARIANCE respectively will be used

Inventory Revaluation

Enable any revaluation of Stock to be recorded

Debit	Credit
-------	--------

*(Warehouse) <i>STOCK</i>	<i>REVALUATION OF STOCK VARIANCE</i>
---------------------------	--------------------------------------

* If the Warehouse does not contain a Cost Centre against INVENTORY or RE-VALUATION then the default Cost Centre matched to STOCK or REVALUATION OF STOCK VARIANCE respectively will be used

POS Station Withdrawals/Receipts

Record the Withdrawal or Receipt of miscellaneous cash from the POS Station

Debit	Credit
<i>POS EXPENSES</i>	<i>UNDEPOSITED FUNDS</i>

POS End of Day Variations

Record variation in actual -v- calculated End of Day Cash-up

Debit	Credit
<i>UNDEPOSITED FUNDS</i>	<i>POS END OF DAY</i>

19.3.7 Advanced Cost Centre Mapping

There may be instances where you wish to look deeper in assessing costs within the following areas

- Sales
- Labour
- Inventory

Ostendo will see if the specific record appears here before going to the Base Mapping described in Section 5.

1. Sales Mapping

A Sales Mapping screen extends the base functionality by allowing you to establish a more detailed structure specifically covering the Sales area. Go to **File>Financial Configuration>Sales Mapping Matrix**. This enables you to define Income and COG Cost Centres for a variety of combinations from.

- 1 - Invoice Customer Type
- 2 - Invoice Customer Region
- 3 - Order Customer Type
- 4 - Order Customer Region
- 5 - Order Class
- 6 - Order Type
- 7 - Sales Order Person
- 8 - Category

In this matrix the level 8 (Category) takes the most significant whereas Level 1 (Invoice Customer Type) takes the least significant. For example:

If there are three mapping records containing:

- Category/Order Type/Order Class/Order Customer Type (I.e. Levels 8/6/5/3)
- Category/Sales Order Person/Order Customer Region (I.e. Levels 8/7/4)
- Sales Order Person/Order Class/Invoice Customer Region (I.e. Levels 7/5/2)

If a Sales record had matching fields defined in all three mapping records then the second mapping record would take priority over the others.

If the Sales record had matching fields except for Category defined in all three mapping records then the third mapping record would take priority over the others.

Let's do an example with this Sales Matrix. Say that we wish to segregate Sales by Job Type. The process would be as follows:

Go into **General>Cost Centres** and create one Cost Centre to represent **Income** and another to represent **COGS** for this Job Type

Now go to **Configuration>Sales Mapping Matrix** and click the 'Add' button to create a new line then go to field 'Order Class' and select 'Job Orders' from the drop-down list. Now go to field 'Order Type' and select 'Progress'. Finally go to fields 'Income Cost Centre' and 'Cost of Goods Cost Centre' and select the Cost Centres that you have just created.

All Job Order with Order Type 'Progress' will now be posted to these Cost Centres. Create a Job Order then process the Invoice as described in the Job Orders training Guide then go into **General>Reports** and select 'Financial Batch Reports'. Enter the following parameters:

- Transaction Status:** Select 'Ready to Send'
- From Transaction Date:** Select Today's Date
- To Transaction Date:** Select Today's Date
- From Batch No:** Leave Blank
- To Batch No:** Leave Blank
- Exclude Conditions:** 'Check' this checkbox

Click the **OK** button bring the report back to the screen. You will see that the generated Journals now use the Job Order specific Cost Centres

2. Labour Mapping

As with Sales Mapping the Labour Mapping screen extends the base functionality by allowing you to establish a more detailed structure specifically covering Labour activities. Go to **File>Financial Configuration>Labour Mapping Matrix**. This enables you to define Direct Labour, Fixed OH, and Variable O/H Cost Centres for a variety of combinations from.

This screen extends that basic functionality and allows a User to establish a more detailed structure covering the Labour Code area. It consists of a hierarchical structure using

- 1 - Category
- 2 - Labour Code Department
- 3 - Employee Department
- 4 - Labour Code
- 5 - Employee

And mapping them to

- Direct Labour Cost Centre
- Fixed OH Cost Centre
- Variable OH Cost Centre

In the above matrix the level 5 (Employee) takes the most significant whereas Level 1 (Category) takes the least significant. For example:

If there are two mapping records containing:

- Employee/Employee Department/Category (I.e. Levels 5/3/1)
- Employee/Labour Code/Category (I.e. Levels 5/4/1)

The second mapping record would take priority over the first.

3. Inventory Mapping

If you wish to segregate Stock and Stock Movements by Warehouse then you can specifically define Cost Centres to be used by each Warehouse. If the Warehouse is not given any specific Cost centres then the system defaults will apply..

Go to **Inventory>Warehouses** and define Cost centres to cover the following:

Inventory: All Stock in this Warehouse will be posted to this Cost Centre.

Adjustments: Stock Adjustments against this Warehouse are 'received from' or 'issued to' this Cost Centre.

Negative Stock: Stock adjustment variances can occur whenever you receive Stock but the current stock levels are negative. In this case the Average Cost for the Item is not amended but the difference between the Average Cost and the Cost of this receipt is 'posted' to this Cost Centre

Re-Valuation: Whenever the Stock is re-valued (Standard Cost or Average Cost) the difference between the old value and the new value is 'posted' to this Cost Centre

Stock Count: Stock variances (plus or minus) as a result of a Stock Count are 'posted' to this Cost Centre.

Receipt Cost: Used if the Inventory Costing method is 'Standard Costing' and this stores the cost difference between the Assembly Order actual Cost and the Standard Cost of the Item.
