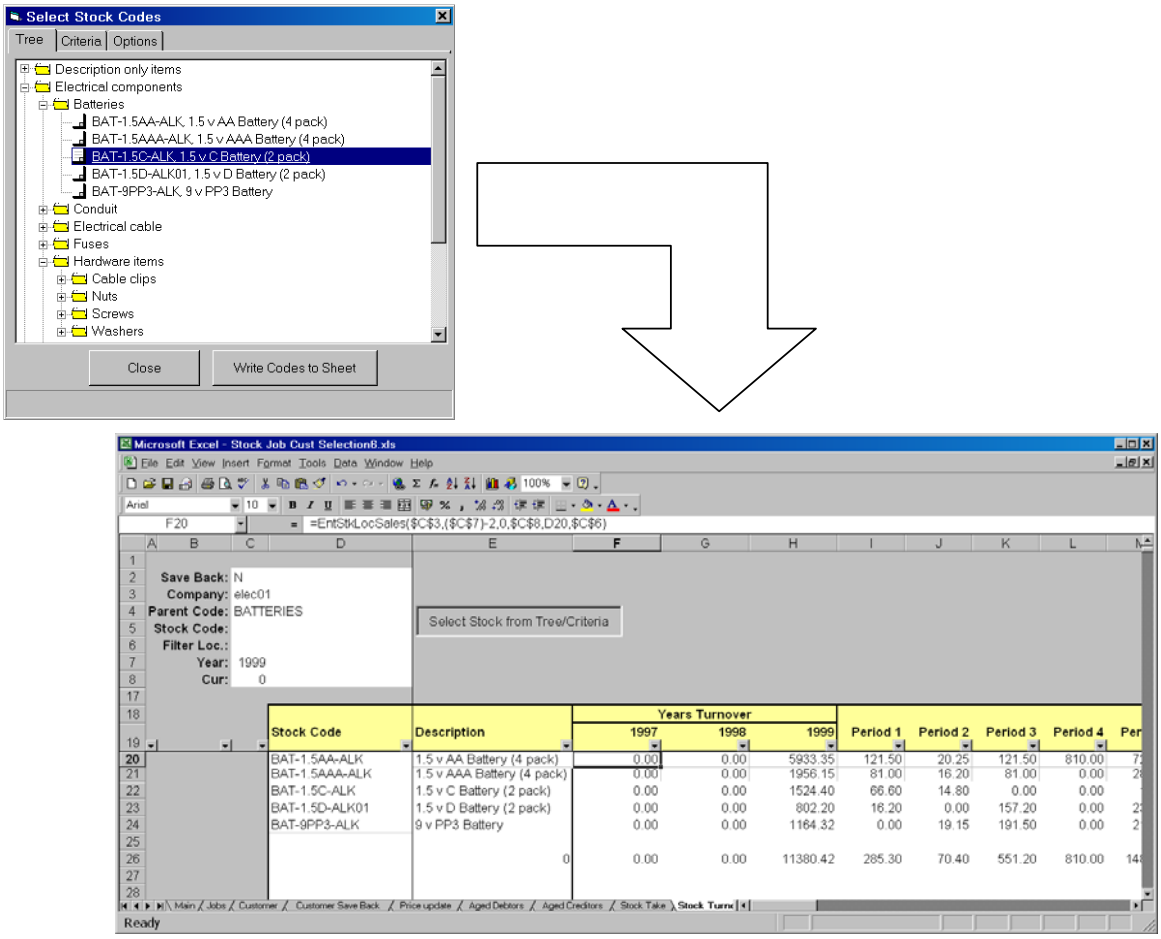
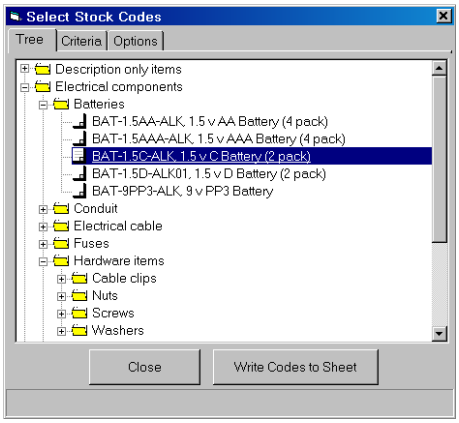
ComTek Functions Version 1.0.0



What is ComTek Functions?



ComTek Functions is an ActiveX that is similar to Exchequer's OLE EntFuncs with which we are all familiar. It has been written to enhance the EntFuncs functionality in Excel by enabling the user to easily retrieve the ID codes (Customer/Supplier, Stock and Job) used by EntFuncs to retrieve and manipulate Exchequer Enterprise data.

The ActiveX will display a floating window for selection from a tree or criteria. Once the required data has been selected the ActiveX writes the ID codes to the spreadsheet and copies down any formulae in adjacent rows (See page 7).

The demo spreadsheet demonstrates the functions provided by the ComTek Functions ActiveX from ComTek Accounts. In the future extra functionality may be added.

Uses

ComTek Functions has many uses to enhance the EntFuncs usability. It is especially useful for Stock Take or Price updates using OLE Save functions.

Please test your requirements before selling this control. ComTek Accounts will not be held responsible! Support on changes/problems, which are not bugs in the ActiveX, may be chargeable.

Requirements

The same as EntFuncs, plus Exchequer V4.32+ is loaded with the COM Toolkit enabled and registered on each PC.

Installation

ComTek Functions is supplied in a ZIP file. Please extract these files to a directory before use. Then copy the ComtekFuncs.dll to the Exchequer Enterprise directory.

Open the example spreadsheet. It may report you need to update external reference to another spreadsheet. Answer NO.

When Excel has finished loading the sheet for the first time, press the [Fix OLE References] button. This will remove any bad references to EntFuncs.

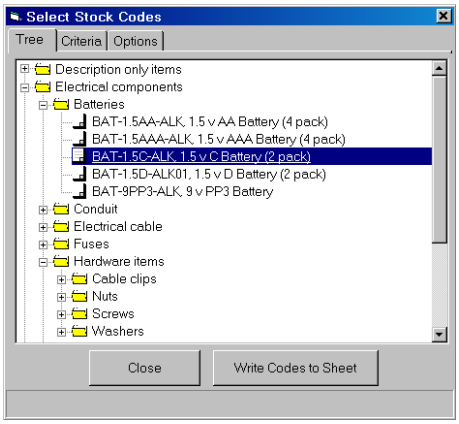
Now enter a valid OLE code in cell I2 on the "Main" worksheet. All other sheets in the workbook reference this cell for the OLE code. EntFuncs will now ask you to log into Enterprise and extract data from your company.

Note the example uses all of the Enterprise OLE Add-ins, so please make sure you have them enabled (they are only required to demonstrate all of the functions).

Examples

Stock codes from tree

Now go to the “Stock Turnover” sheet. Press [Select Stock from Tree/Criteria] button. A splash screen will appear to inform you which company you are connected to. Press [OK] button.



A floating window should now appear:

Select a section such as “Batteries” in the tree and press [Write Codes to Sheet]. The ActiveX will then write the stock codes to cell D20 down. After writing the codes it will copy any functions found in row 20 (in this case OLE functions to return the Stock Turnover figures).

After this it will:

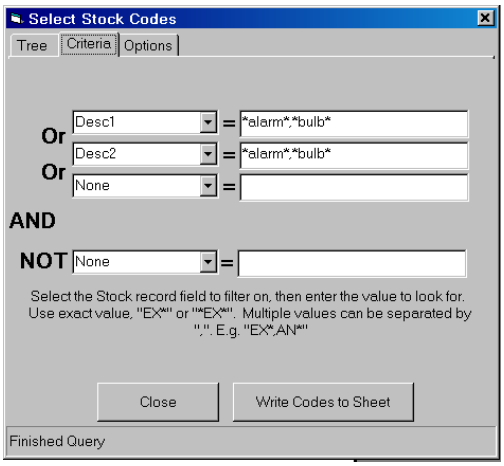
1. Set the print area
2. Optionally put a total at the bottom of the data
3. Move the cursor to a defined position

At this point you can close the floating window or select different data. For example selecting the heading “Electrical components” will return all codes under heading (with the exclusion of Heading codes).

Stock codes by criteria

This time select the “Stock Take” or “Price update” spreadsheet.

Press [Select Stock from Tree/Criteria] button to display floating window and select the “Criteria” tab. Here you can specify criteria to filter the stock records.



In the example we have asked for Stock codes where the stock record contains the word alarm or bulb in Description line 1 or 2.

This allows you to select on most of the stock fields, so we could select stock records where User1 = “LIGHTS”,”FUSES” or User1 = “STARTWITH*”,”*CONTAINS*”, etc.

So we could perform a stock take or price update from a specific Supplier, Area, Location, or User defined section of the stock tree.

Note: The “Options” tab allows you to hide BOM, Description, etc. items

The above functionality is also available for Jobs and Customer/Supplier records - please explore!

How does it work?

Each sheet has a button on it that activates the ActiveX to retrieve data. The ActiveX requires many parameters to alter its behaviour. These parameters basically tell the ActiveX:

1. OLE code for company
2. Where to put the selected data
3. Define the top/width of print area
4. Where to leave the cursor after query
5. Recurse tree ON/OFF (allows the user to select Tree Heading and fill the sheet with all sub items)
6. Apply auto filter ON/OFF
7. Put sum at bottom of data ON/OFF.

Show Tree/Criteria button

To view this code, go to the “Price update” spreadsheet. Right click on Excels Button bar and select “Control Toolbox” Toolbar. Now press button this will put Excel in the control edit mode. Now double click on the [Select Stock from Tree/Criteria] button. Excel will now switch to the Visual Basic editor.

The code should read:

```
Private Sub CommandButton1_Click()
    'Call Display tree to allow selection of stock
    'Parameters
    '   Company Code cell ref
    '   Column to write back codes to
    '   Data row start where functions to copy down spreadsheet are
    '   Parent Stock code cell ref
    '   Stock code Cell ref
    '   Location code cell ref
    '   Top left hand cornet of Print area
    '   Right most column of Print area
    '   Move cursor to after query
    '   Recurse the tree when getting the Stock Codes (if true will
return all stock codes under any heading)
    '   Apply Auto Filter after query
    '   Sum at bottom
    '   Copy Formula/Values from row (used to reset cell which are
overwritten by user)

    ZZ_DisplayStockTree "C3", "D", 20, "C4", "C5", "C6", "D18", "O",
"K22", True, False, False, 15

End Sub
```

Basically this button calls a function called `ZZ_DisplayStockTree` that expects to find the OLE code in cell C3. It will write the codes back to column "D" from row 20 onwards. As the user selects parts of the stock from the tree it will write the current parent code to cell C4 and stock code to C5. If Locations are turned on the selected location will be written to C6.

After writing the data back then select the print area with cell D18 as the top left hand co-ordinate, and column "O" as the right hand most edge of area. The bottom will be set to the last row written.

After writing the codes moves the cursor to cell K22. This is useful for the update sheets, as you can move the cursor to the start position.

Allow recursion of tree. If True then selecting a heading in the tree will write back all codes under that heading and any other headings below it. If False will just return those codes which belong to that heading.

Apply auto filter after writing data back True/False. This allows the user to then select a particular supplier, top 10 or customer criteria. Any totals will reflect these filters.

Write Sum formula at bottom of data True/False.

Optional: Copy Formulae/Values from row 15. This is very important for the Update spreadsheets.

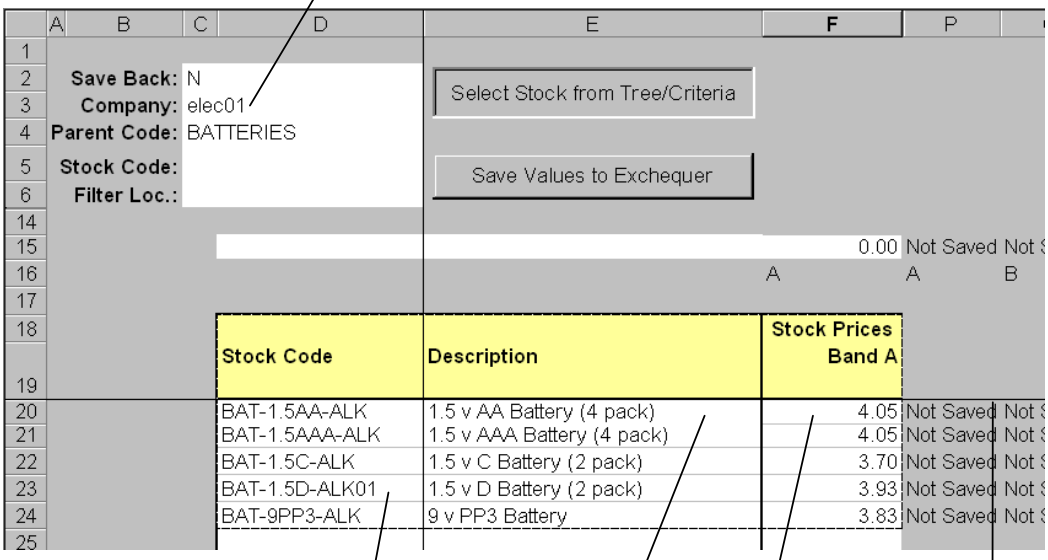
Save Back button

The [Save values back to Exchequer] button asks the user if they want to update the values in Exchequer. It simply changes the value of cell C2 from "N" to "Y". The normal formula for saving is then used:

```
=IF($C$2="Y",EntStkSaveSalesPrice($C$3,$D20,P$16,F20),"Not Saved")
```

Therefore in this example we have:

OLE Code C3
Parent/Stock/Location codes written C4-6



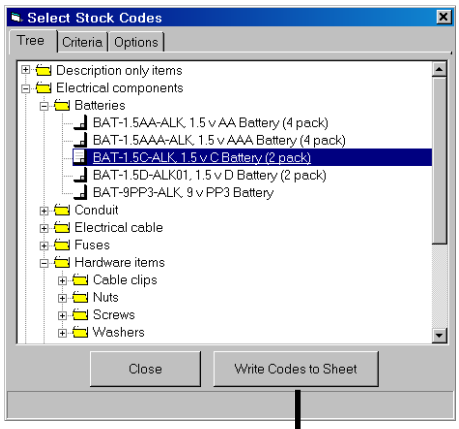
Stock codes written to D20 and down

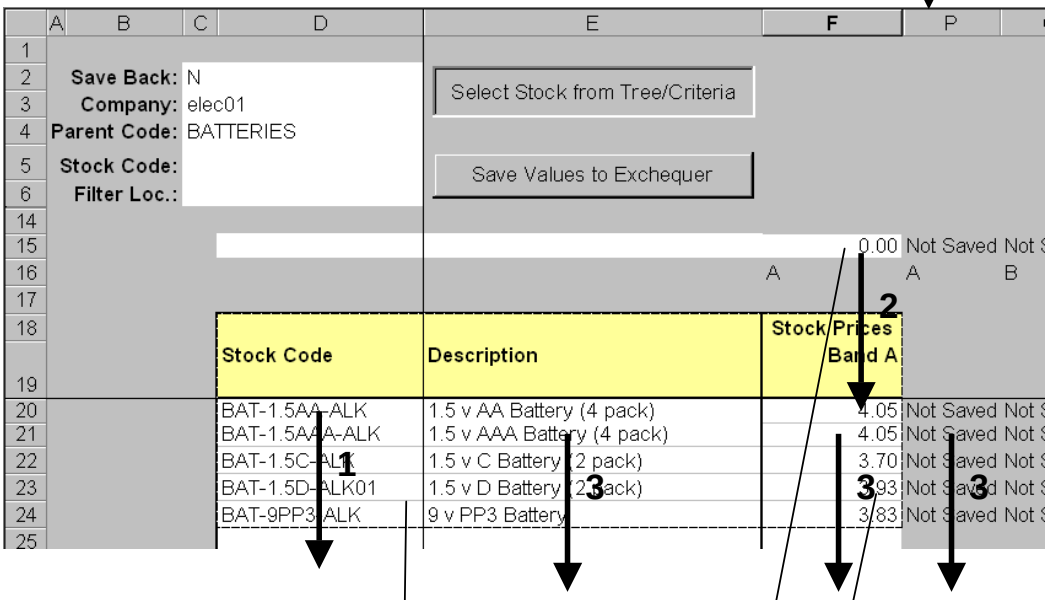
OLE function EntStkDescr to get stock Description
=EntStkDescr(C3,D20,1)
=EntStkDescr("ELEC01","BAT-1.5AA-ALK",1)

OLE function EntStkLocSalesPrice to get Price for Band A
=EntStkLocSalesPrice(C3,$D20,$C$6,F$16)
=EntStkLocSalesPrice("ELEC01","BAT-1.5AA-ALK","","A")
User enters new values in this area, so restores formulae by copying formula/values from row 15 to 20. See next page.

OLE function EntStkSaveSalesPrice to save Price for Band A when button is pressed
Cell C2 changes to "Y"
=IF(C2="Y",EntStkSaveSalesPrice(C3,$D20,P$16,F20),"Not Saved")
=IF(C2="Y",EntStkSaveSalesPrice("ELEC01","BAT-1.5AA-ALK","A",NewValue),"Not Saved")

The process after a Query:





1) Stock codes written to D20 and down

2) Because this sheet is used for updating values in Exchequer the user types over the formulas in cells F20 down. To restore these formulas ComTek Functions copies any formula/values from row 15 to 20. This is specified in the last parameter of the function. NOTE: it is only used for Save-back sheets. Non save-back sheets do not need this, or perform this step.

3) Values/Formulae in Row 20 are copied down the sheet

4) Sum and Auto filter are applied if turned on

5) Print Area is replied

The ActiveX uses the "Funcs Defaults" spreadsheet to store any defaults settings. DO NOT INSERT OR DELETE ROWS/COLUMNS

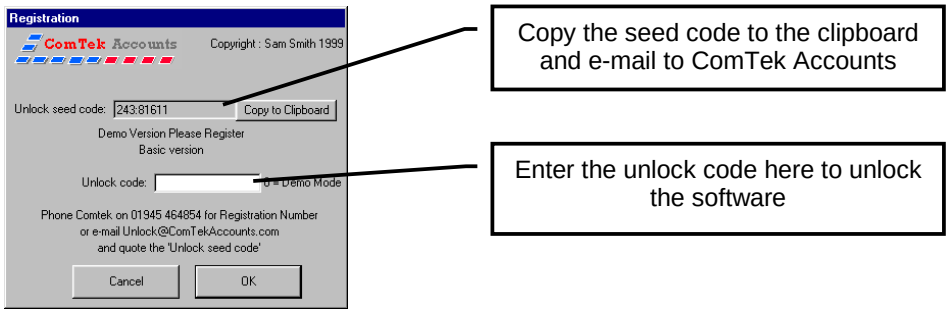
If you want to customise a spread sheet:

1. Check out www.ComTekAccounts.com for any developments
2. Copy the Example spreadsheet
3. Select the sheet which best fits your needs
4. Delete unwanted sheets (must leave in "Funcs Defaults" and it's a good idea to leave "Main" because it has the Reference fix code in it)
5. Customise the sheet, read the next section.
6. Any good examples please e-mail Support@ComTekAccounts.com

Note: The "Stock Shortage" spreadsheet demonstrates a button that applies a filter to the sheet. Customisation such as this can easily be performed by recording macros and running the macro from a button.

Registering ComTek Functions

To register ComTek functions, go to the "Options" menu (on any of the Query windows) and select "Register". The following window will be displayed:



Please e-mail the "Unlock Seed Code" to unlock@ComTekAccounts.com. This seed code will be valid for one week.

Note : Payment must be received, via credit card, before unlock code can be given.

Importing Transactions/Customer/Supplier/Stock/GL records easily

Please have a look at the power of PostTrans for easily importing data into Exchequer from within Excel. It is available on a showcase CD-ROM with many demonstration videos.

Check out http://www.comtekaccounts.com for the latest information