



# Reporting OpenOffice

## Index

|           |   |          |
|-----------|---|----------|
| <b>1.</b> | <b>SET UP OPENOFFICE CONNECTION TO ORCA SERVER.....</b> | <b>2</b> |
| 1.1.      | Install Caché Client.....                               | 2        |
| 1.2.      | Set Up The Caché Connection To The ORCA Server.....     | 2        |
| <b>2.</b> | <b>INSTALLING OR UPGRADING THE TCSADD-IN .....</b>      | <b>2</b> |
| 2.1.      | Install The TCSAdd-In For The First Time .....          | 2        |
| 2.2.      | Upgrade An Existing TCSAdd-In .....                     | 2        |
| <b>3.</b> | <b>TCSADD-IN USAGE .....</b>                            | <b>2</b> |
| 3.1.      | Connecting to ORCA.....                                 | 2        |
| 3.2.      | Budget Load.....  | 3        |
| 3.3.      | Spreadsheet Load.....                                   | 3        |
| 3.4.      | Update Sheet.....                                       | 3        |
| 3.5.      | Connect/Disconnect.....                                 | 3        |
| 3.6.      | ORCA Addin Help .....                                   | 3        |
| 3.7.      | About ORCA .....  | 3        |
| <b>4.</b> | <b>TCS ADD-IN FUNCTIONS.....</b>                        | <b>3</b> |
| 4.1.      | TCSAcctProperty .....                                   | 4        |
| 4.2.      | TCSAcctTotal .....                                      | 5        |
| 4.3.      | TCSDate.....  | 6        |
| 4.4.      | TCSMethod .....   | 7        |
| 4.5.      | TCSProperty .....                                       | 7        |
| 4.6.      | TCSUpdatePrice .....                                    | 8        |
| 4.7.      | TCSCodeProperty .....                                   | 9        |



## 1. SET UP OPENOFFICE CONNECTION TO ORCA SERVER

The OpenOffice TCSAdd-In makes use of Caché direct connectivity, so a 'Caché Client' for ODBC/Caché Direct connection must be installed on the PC.

Prior to starting the installation, obtain the IP address of the ORCA server.

### 1.1. Install Caché Client

- Insert Caché installation CD into CD drive
- Select 'Client' installation type

### 1.2. Set Up The Caché Connection To The ORCA Server

- Right click on the Caché cube
- Navigate to Preferred Server? Add/Edit
- Add the ORCA server if not already there. Make the server name ORCA Live
- Set the ORCA server's IP address
- Delete LOCALTCP server
- Exit Caché Client Manager

## 2. INSTALLING OR UPGRADING THE TCSADD-IN

### 2.1. Install The TCSAdd-In For The First Time

To install the TCSAdd-In on the local PC for the first time, use Explorer to navigate to the location of the OpenOffice Add-In file, TCS-OOAddIn.oxt on your server and double click on the file, alternatively open OpenOffice on the client PC and do the following:

- Using the OpenOffice menu options, choose Tools? Extension Manager...
- Click on the 'Add...' button and navigate to the location of the OpenOffice add-in file, TCS-OOAddIn.oxt on your server.

### 2.2. Upgrade An Existing TCSAdd-In

To upgrade the existing TCSAdd-In on the local PC you will first need to remove the original add-in as follows:

- Open OpenOffice on the client PC and using the OpenOffice menu options, choose Tools? Extension Manager...
- Click on ORCA Addin for OpenOffice
- Click on Remove

Then install the TCSAdd-In upgrade by following the procedure defined in section 2.1 above- Install The TCSAdd-In For The First Time.

## 3. TCSADD-IN USAGE

The usage of the TCSAdd-In is explained in the sections below, with the 1<sup>st</sup> section describing the method of connecting to the ORCA system, and each subsequent section covering one of the drop-down menu options available from the 'ORCA' tab in OpenOffice.

### 3.1. Connecting to ORCA

Note that the ORCA connection is specific to a spreadsheet, if more than one spreadsheet is required to interact with ORCA at the same time, a connection will need to be made from each spreadsheet. A number of the menu options require a connection to the server, in these instances the Connection Manager window will be opened automatically if a connection has not already been established for the current spreadsheet.



To establish a connection manually:

- Open OpenOffice on the client PC.
- Open the required OpenOffice spreadsheet. Using the OpenOffice menu options, choose ORCA? Connect/Disconnect
- Enter the IP address of the ORCA server at the Server Connection prompt, and enter. Note: The last entered IP address will be defaulted the next time a connection to ORCA is made, if a connection is required to a different server, override the defaulted IP address.
- Select the Namespace required, and click on OK.
- A login window will be displayed, enter your ORCA username and password to connect to the server.

### **3.2. Budget Load**

Use this option to load budget data from the current spreadsheet into the ORCA system. It is commonly used for loading multiple journal files and the spreadsheet must adhere to specific format requirements. The benefit of using this option is that a separate journal is created for each specified date. Refer to the [Open Office Example](#) for specific formats and layouts.

### **3.3. Spreadsheet Load**

Use this option to load data from the current spreadsheet into ORCA. It is commonly used for loading journal files and the worksheet must adhere to specific format requirements. The benefit of using this option is that all ORCA validation is performed as the data is loaded, thereby allowing defaults to be set, and preventing invalid data from being loaded. Refer to the [OpenOffice Example](#) for specific formats and layouts.

### **3.4. Update Sheet**

Use this option to refresh data where embedded TCS Functions exist in the current spreadsheet. This will optionally connect to ORCA and re-calculate the data represented by the functions entered as formulas.

### **3.5. Connect/Disconnect**

Use this option to connect to, or disconnect from, the ORCA system:

- If there is no active connection, the connection manager window will be displayed. Follow the procedure in section 3.1 above, in order to connect to ORCA in preparation for subsequent data retrieval or upload.
- If there is an active connection when the Connect/Disconnect option is selected, it will be disconnected.

### **3.6. ORCA Addin Help**

Use this option to view this ORCA AddIn procedure document on the client PC.

### **3.7. About ORCA**

Use this option for support purposes. It contains the current version number of the ORCA system and also allows (where possible) for the automatic creation of support email including the current workbook as an example.

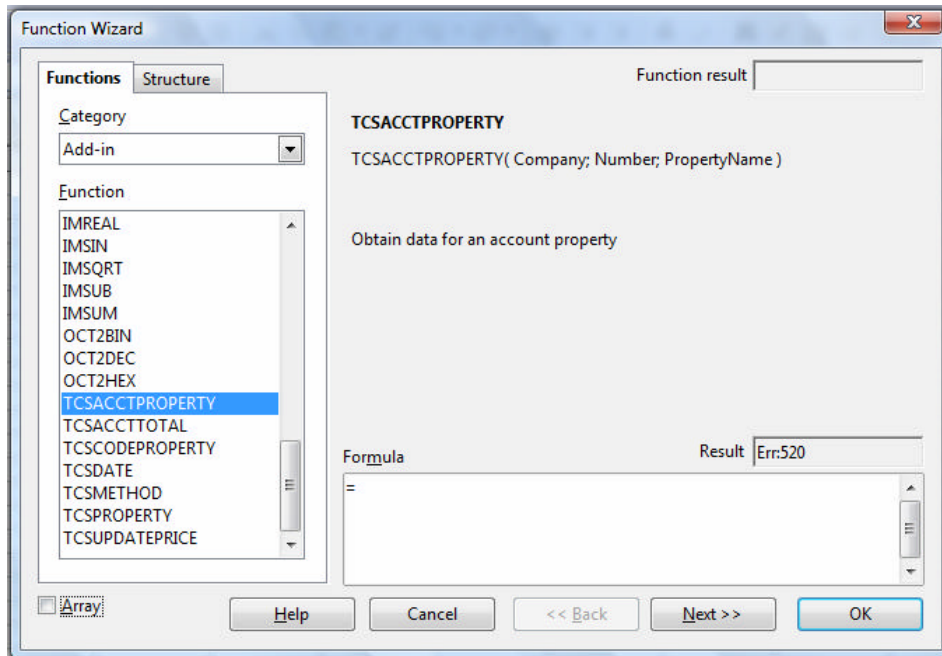
## **4. TCS ADD-IN FUNCTIONS**

Access the function wizard and select the Add-In category in order to access the TCS functions that are available to users of the TCSAdd-In. Select the required TCSAdd-In function, and proceed to the Structure form in order to input the function arguments.

Note that when entering function arguments, the following input rules apply:

- Cell references can be entered as arguments.
- Data values entered directly as function arguments, including dates, property and method names, must be entered in double quotes, e.g. "Name" or "25/6/2009".

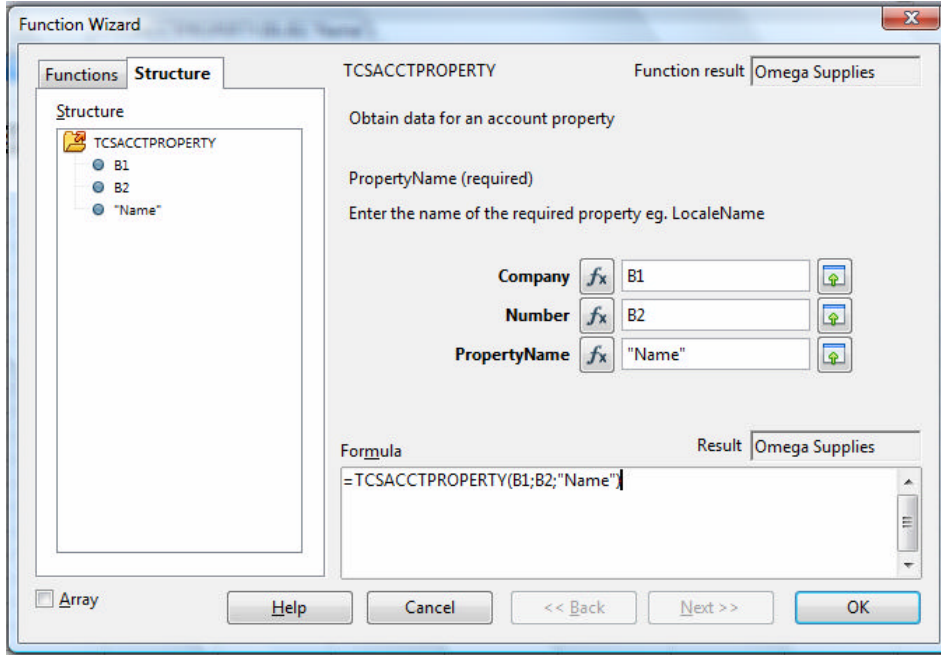
Refer to the [OpenOffice Example](#) for an example of each function.



#### 4.1. TCSAcctProperty

This function is used to return the value of a selected property (or attribute) of the TCSAcct class instance. This is achieved by passing a valid company and account number as actual inputs, or using the relevant cell references, as well as a valid ORCA object class property name. It requires 3 input arguments, in the following order:

1. Company - ORCA Company Code
2. Number - ORCA Account Number
3. PropertyName - ORCA object class property name required



## 4.2. TCSAcctTotal

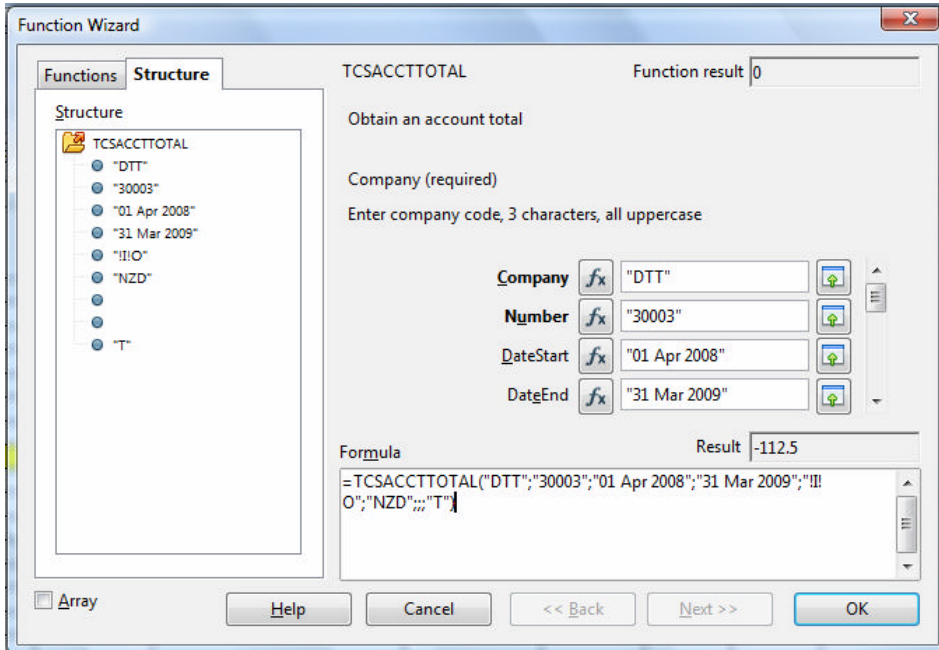
This function is used to return a total of Amount or Quantity as a numeric value from a given account. The account may be any valid ORCA account and optionally passed additional filters.

The function requires 2 input arguments followed by optional parameters in the following order:

- |   |                 |
|---|-----------------|
| 1. Company - ORCA Company Code.                                   | Default         |
| 2. Number - ORCA Account Number.                                  |                 |
| 3. Optional Start Date for return total - DateStart               | Beginning       |
| 4. Optional End Date for return total - DateEnd                   | End             |
| 5. Optional "!" Separated list of transaction Codes - TranCodes   | NULL for ALL    |
| 6. Optional Foreign Exchange Code - FXCode                        | Company FX      |
| 7. Optional Financial Transaction Type (0/1/2/3) - FinancialType  | 0 for Financial |
| 8. Optional Suppress Year End transactions flag(0/1) - YearEnd    | 0 for NO        |
| 9. Optional Return Total or Quantity (T/Q) - TotalorQty           | T for Total     |
| 10. Optional Include Year to date only values flag(0/1) - YTDOnly | 0 for NO        |
| 11. Optional Analysis filter - AnalysisNumber                     | NULL for ALL    |
| 12. Optional Return Cash transactions only flag(0/1) - CashTrans  | 0 for NO        |

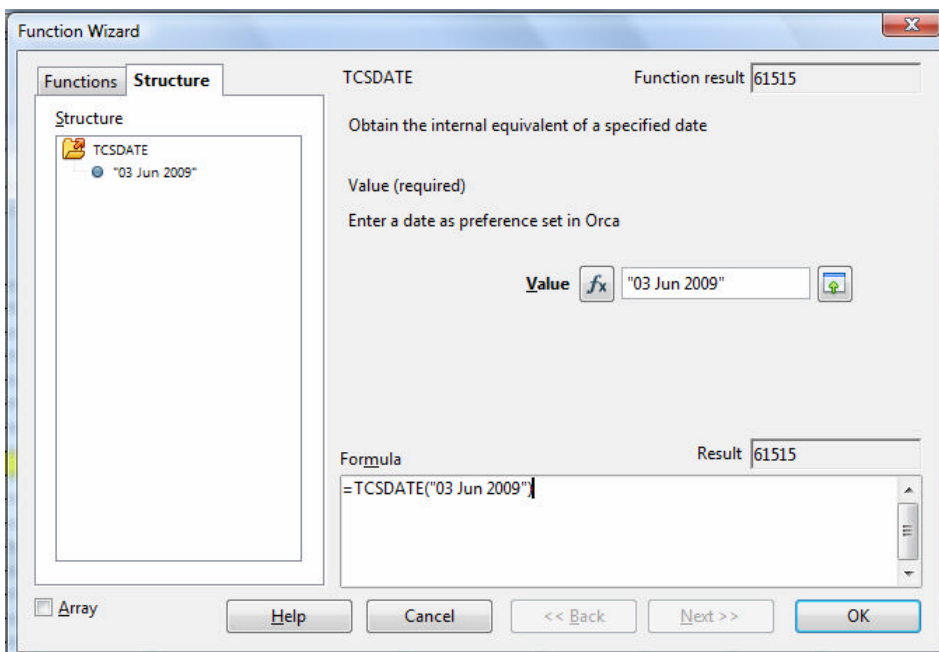
E.g.

|             |   |            |
|-------------|---|------------|
| Argument 1  | = | DTT        |
| Argument 2  | = | 30003      |
| Argument 3  | = | 01/01/2002 |
| Argument 4  | = | 31/12/2003 |
| Argument 5  | = | !!O        |
| Argument 6  | = | NZD        |
| Argument 7  | = | 0          |
| Argument 8  | = | 0          |
| Argument 9  | = | Q          |
| Argument 10 | = | 0          |
| Argument 11 | = |            |
| Argument 12 | = | 1          |



### 4.3. TCSDate

This function is used where external date values must be converted to the internal format understood by the ORCA. system. The internal format is a positive integer containing the number of days since 31 December 1840. The function requires 1 input argument, being the external date value, e.g. "03 Jun 2009" input into the function will be returned as 61515. The date format is in accordance to preference set in ORCA. If ORCA is set to "default", date format is accordance to regional settings of the client PC.



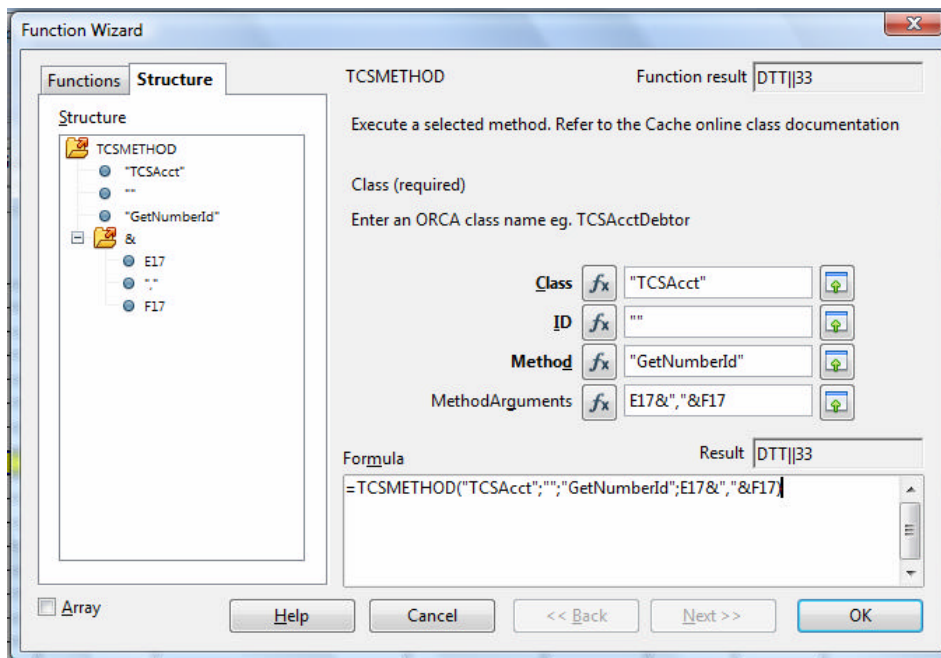
#### 4.4. TCSMethod

This function is used to execute public ORCA object methods. A commonly used example is to return the identifier of an account object for subsequent use as input into the TCSPROPERTY or TCSUPDATEPRICE functions. It requires 4 input arguments, in the following order:

1. ORCA object class name.
2. ORCA object class instance identifier (may be blank).
3. ORCA object class method name.
4. Method arguments, as a single string of values delimited by commas.

E.g. Argument 1 = "TCSAcct"  
 Argument 2 = ""  
 Argument 3 = "GetNumberId"  
 Argument 4 = "E17&","&F17"

Note that argument 4 contains 2 spreadsheet cell addresses concatenated together with a comma between them. This forms a string comprising of "company code,account number", e.g. "DTT,30003", and will return the ID for the account, e.g. DTT|33.



#### 4.5. TCSPROPERTY

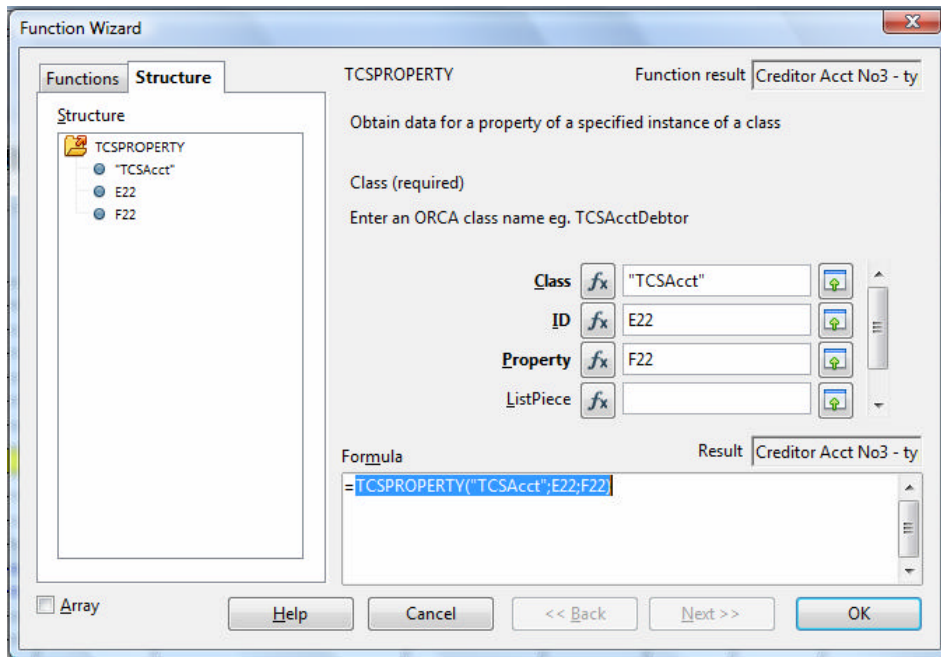
This function is used to return the value of a selected property (or attribute) of the specified class instance. This is different from TCSMETHOD, which returns the execution results of a selected method of the class instance. This is achieved by passing any valid instance identifier and property combination. It requires 5 input arguments, in the following order:

1. ORCA object class name.
2. ORCA object class instance identifier.
3. ORCA object class property name.
4. List Piece number (may be blank). This is an integer that is used to determine which piece of data in an ORCA serial object is to be returned. If it is left blank the function will return all the data unchanged.
5. Logical Flag (may be blank). This is a boolean value that is used to determine the format in which the data is to be returned. If Logical Flag is set to 1, or is left blank, then the data is returned in ORCA internal format (i.e. logical format), while if Logical

Flag is 0 then it is in display format. This is particularly useful for dates – refer to function [TCSDate](#).

E.g. Argument 1 = "TCSAcct"  
 Argument 2 = E22  
 Argument 3 = F22  
 Argument 4 =  
 Argument 5 = 1

Note that arguments 3 & 4 each contain a spreadsheet cell address. If those addresses contain the data "DTT||33" and "LocaleName" respectively, the function returns "Creditor Acct No3 – type C".



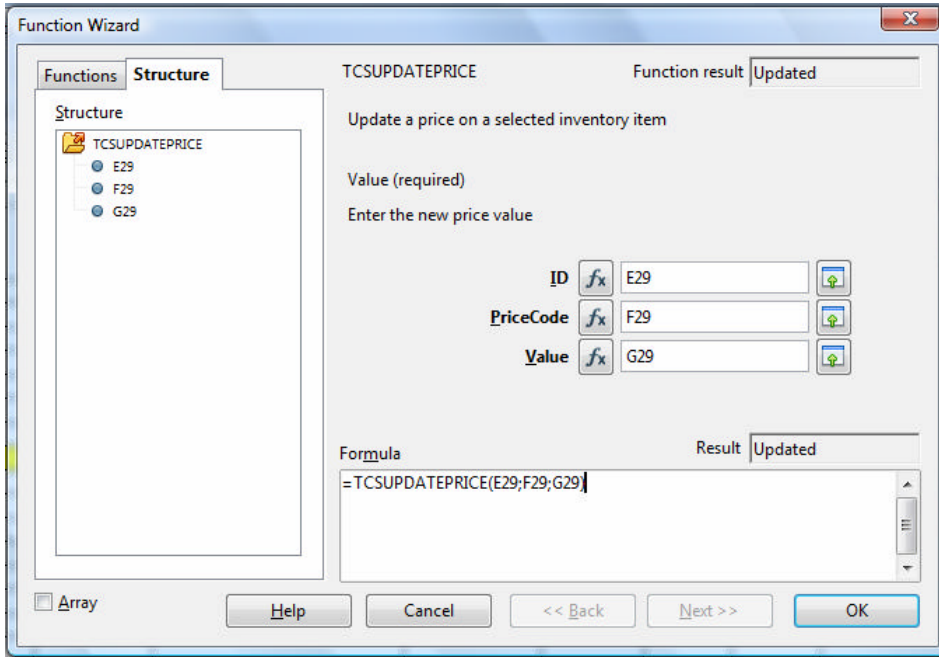
#### 4.6. TCSUpdatePrice

This function is used to update a given price code on an inventory item with the specified value. If the price was updated successfully, then the string "Updated" is returned, otherwise "Failed" is returned. Note that here 'inventory item' means an instance of the ORCA class TCSAcctInventory. The function requires 3 input arguments in the following order:

1. ORCA object class instance identifier.
2. ORCA price code
3. New price value.

E.g. Argument 1 = E29  
 Argument 2 = F29  
 Argument 3 = G29

Note that all arguments each contain a spreadsheet cell address. If those addresses contain the data "DTT|41", "E" and "19.55" respectively, the function returns "Updated", signifying that the item was successfully updated with the new price.



#### 4.7. TCSCodeProperty

This function is used to return the value of a selected property (or attribute) of the TCSCode class instance. This is achieved by passing any valid company and account number, any valid code and a valid ORCA class property name. It requires 4 input arguments, in the following order:

1. Company - ORCA Company Code.
2. CodeType - This will be concatenated with "TCSCode" to form the class instance
3. Code – Code as listed in the ORCA Company.
4. ORCA object class property name.

E.g. Argument 1 = NPM  
 Argument 2 = FX – Will be converted to TCSCodeFX  
 Argument 3 = USD  
 Argument 4 = LocaleName

Function Wizard

Functions **Structure**

Structure

- TCSCODEPROPERTY
  - "DTT"
  - "FX"
  - "USD"
  - "LocaleName"

TCSCODEPROPERTY      Function result: United States Dollar

Obtain data for a code property

PropertyName (required)  
Enter the name of the required property eg. LocaleName

Company

CodeType

Code

PropertyName

Formula      Result: United States Dollar

=TCSCODEPROPERTY("DTT";"FX";"USD";"LocaleName")

Array