



# Reporting MS Excel

## Index

<b>1.</b>	<b>SET UP EXCEL CONNECTION TO ORCA SERVER</b>	<b>2</b>
1.1.	Install Caché Client	2
1.2.	Set Up The Caché Connection To The O.R.C.A Server	2
1.3.	Set Up Excel Connection To O.R.C.A 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	3
<b>3.</b>	<b>TCSADD-IN USAGE</b>	<b>3</b>
3.1.	Connecting to The O.R.C.A System	3
3.2.	Budget Load	4
3.3.	Spreadsheet Load	4
3.4.	Update Sheet	4
3.5.	Connect	4
3.6.	Disconnect	4
3.7.	About O.R.C.A	4
<b>4.</b>	<b>TCS ADD-IN FUNCTIONS</b>	<b>4</b>
4.1.	TCSDate	5
4.2.	TCSMethod	5
4.3.	TCSProperty	6
4.4.	TCSUpdatePrice	7
4.5.	TCSAcctTotal	7
4.6.	TCSAcctProperty	8
4.7.	TCSCodeProperty	8



## 1. SET UP EXCEL CONNECTION TO ORCA SERVER

The Excel 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 O.R.C.A 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 O.R.C.A Server

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

### 1.3. Set Up Excel Connection To O.R.C.A Server

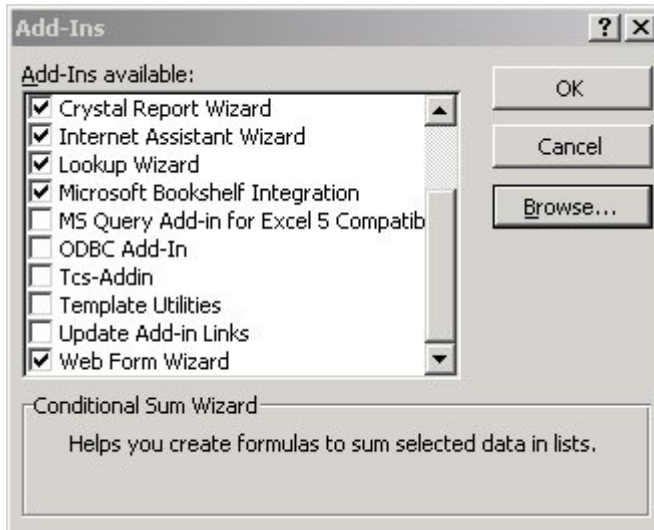
- Open MS Excel on the client PC.
- Using the Excel menu options, choose O.R.C.A→Connect→New...
- Enter ORCA Live into the Connection Name
- Enter the O.R.C.A server's IP address into Server
- Enter the company namespace (e.g. IZZ) into Namespace
- Confirm the the Port is set to 1972
- Left click OK
- Left click Connect (to test the connection)
- Enter user name and password

## 2. INSTALLING OR UPGRADING THE TCSADD-IN

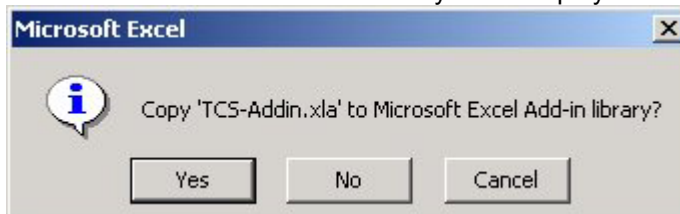
### 2.1. Install The TCSAdd-in For The First Time

To install the TCSAdd-in on the local PC for the first time, execute the following tasks in order:

- Open MS Excel on the client PC.
- Using the Excel menu options, choose Tools→Add-ins...
- Left click the 'Browse' button and navigate to the location of the Excel add-in file, TCSAdd-in.xla on your server.



- Double-click or choose the file above to Open.
- If installation is from a separate server, the message “Copy ‘TCSAdd-in.xla’ to Microsoft Excel Add-in library?” will display.



- Left click Yes.
- Cancel the Add-in form. At the top of the Excel window, there should now be an additional menu option called O.R.C.A, for use with all spreadsheets. Also supplied with the add-in are ‘User defined Functions’ for using the data retrieval methods.

## 2.2. Upgrade An Existing TCSAdd-in

To upgrade the existing TCSAdd-in on the local PC, execute the following tasks in order:

- Open MS Excel on the client PC.
- Using the Excel menu options, choose Tools→Add-ins...
- Left click the ‘Browse’ button and navigate to the location of the excel add-in file, TCSAdd-in.xla on your server.
- Double-click or choose the file above to Open and the message “Copy ‘TCSAdd-in.xla’ to Microsoft Excel Add-in library?” will display if installation is from a separate server.
- Left click Yes.
- A message appears stating that “A file named ‘TCSAdd-in.xla’ already exists in this location. Do you want to replace it?”. Left click Yes.
- Cancel the Add-in form.

## 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 O.R.C.A system, and each subsequent section covering one of the drop-down menu options available from the ‘O.R.C.A’ tab in Excel.

### 3.1. Connecting to The O.R.C.A System

- Open MS Excel on the client PC.
- Using the Excel menu options, choose O.R.C.A→Connect



- If there is no connection visible to the O.R.C.A server, create one (refer to [Set Up Excel Connection To The O.R.C.A Server](#)).
- Highlight the required connection
- Left click Connect
- Enter user name and password into the login window.

### **3.2. Budget Load**

Use this option to load budget data from the current spreadsheet into the O.R.C.A system. It is commonly used for loading multiple journal files and 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 [Excel Example](#) for specific formats and layouts.

### **3.3. Spreadsheet Load**

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

### **3.4. Update Sheet**

Use this option to refresh data where embedded TCS Functions exist in the current worksheet. This will optionally connect to O.R.C.A and re-calculate the data provided by the functions entered as formulas.

### **3.5. Connect**

Use this option to connect to the O.R.C.A system in preparation for subsequent data retrieval or upload. This ensures a connection is available when required.

### **3.6. Disconnect**

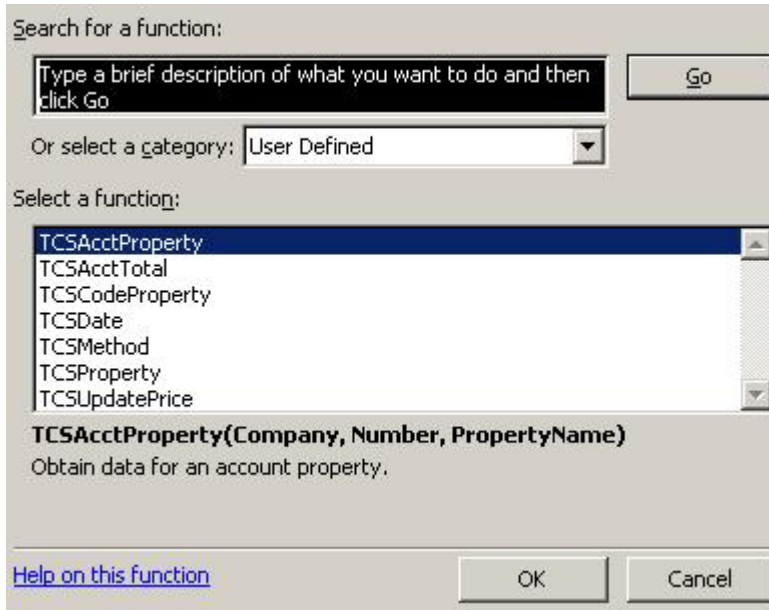
Use this option to disconnect from the O.R.C.A system, thereby freeing up resources and possibly a licence unit for others to use.

### **3.7. About O.R.C.A**

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

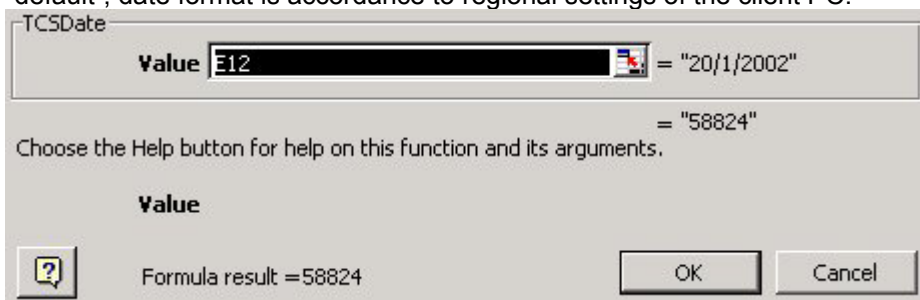
## **4. TCS ADD-IN FUNCTIONS**

The following functions are available to users of the TCSAdd-in. Refer to [Excel Example](#), or use the following link for an example of each function.



#### 4.1. TCSDate

This function is used where external date values must be converted to the internal format understood by the O.R.C.A. 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. "02 Aug 2002" input into the function will be returned as 59018. 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.



#### 4.2. TCSMethod

This function is used to execute public O.R.C.A 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. O.R.C.A object class name.
2. O.R.C.A object class instance identifier (may be blank).
3. O.R.C.A 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. If they form the string "DTT,60001", then the function returns "DTT|33".

TCSMethod

Class "TCSAcct" = "TCSAcct"

ID "" = ""

Method "GetNumberId" = "GetNumberId"

MethodArguments "E17&","&F17" = "DTT,60001"

Formula result = "DTT||33"

Choose the Help button for help on this function and its arguments.

Class

Formula result =DTT||33

OK Cancel

### 4.3. 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. O.R.C.A object class name.
2. O.R.C.A object class instance identifier.
3. O.R.C.A 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 O.R.C.A 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 O.R.C.A 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 "Inventory Acct No1 – type I".

TCSProperty

Class "TCSAcct" = "TCSAcct"

ID "E22" = "DTT||33"

Property "F22" = "LocaleName"

ListPiece "" = ""

Logical "1" = 1

Formula result = "Inventory Acct No1 - type I"

Choose the Help button for help on this function and its arguments.

Class

Formula result =Inventory Acct No1 - type

OK Cancel

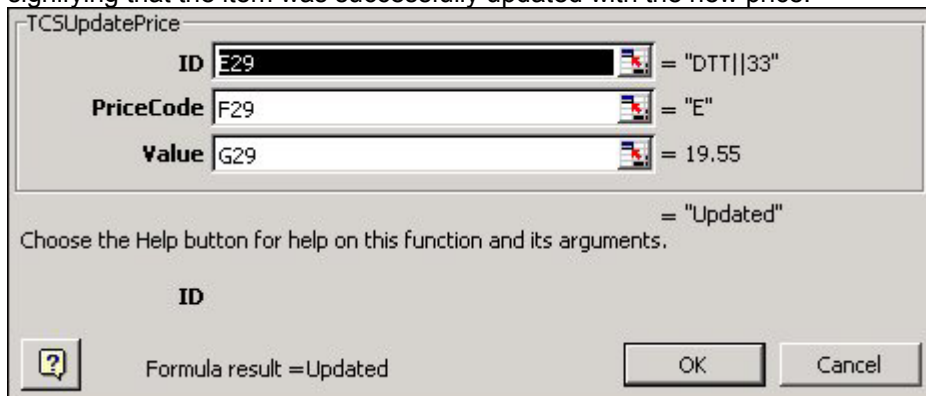
#### 4.4. TCSUpdatePrice

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 O.R.C.A class TCSAcctInventory. The function requires 3 input arguments in the following order:

1. O.R.C.A object class instance identifier.
2. O.R.C.A 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||33", "E" and "19.55" respectively, the function returns "Updated", signifying that the item was successfully updated with the new price.



#### 4.5. 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               |
| 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

TCSAcctTotal

Company  = "DTT"

Number  = 30003

DateStart  =

DateEnd  =

TranCodes  =

= -225

Choose the Help button for help on this function and its arguments.

**Company**

Formula result = -225

#### 4.6. 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 any valid company and account number, 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. ORCA object class property name.

E.g. Argument 1 = NPM  
 Argument 2 = D0100  
 Argument 3 = Name

TCSAcctProperty

Company  = "NPM"

Number  = "D0100"

PropertyName  = "Name"

= "David Jones - HO"

Obtain data for an account property.

**Company** Enter company code, 3 characters, all uppercase.

#### 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. O.R.C.A object class property name.



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

TCSCodeProperty

Company	<input type="text" value="NPM"/>		= "NPM"
CodeType	<input type="text" value="FX"/>		= "FX"
Code	<input type="text" value="USD"/>		= "USD"
PropertyName	<input type="text" value="LocaleName"/>		= "LocaleName"

Obtain data for a code property .

**PropertyName** Enter the name of the required property eg. LocaleName

= "United States Dollar"